



Design Document

CA SCR 205172 CIV SCR 102502 – Increase the threshold for CalWORKs Overpayments Phase 1

CalACES	DOCUMENT APPROVAL HISTORY	
	Prepared By	Sidhant Garg
	Reviewed By	Kapil Santosh, Jyothirmayi Chavata

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
02/28/2018	.01	Initial Revision	Sidhant Garg

APPROVAL DATE	APPROVED VERSION	APPROVER

DRAFT

Table of Contents

1	Overview	5
1.1	Current Design.....	5
1.2	Request	5
1.3	Overview of Recommendations	5
1.4	Assumptions	5
2	Recommendations	6
2.1	Recovery Account Detail	6
2.1.1	Overview.....	6
2.1.2	Description of Changes.....	6
2.1.3	Page Location	6
2.1.4	Counties Impacted	6
2.1.5	Security Updates.....	6
2.1.6	Page Mapping.....	7
2.2	Uncollectible Recovery Account Batch.....	7
2.2.1	Overview.....	7
2.2.2	Description of Changes.....	7
2.2.3	Execution Frequency	7
2.2.4	Key Scheduling Dependencies	7
2.2.5	Counties Impacted	8
2.2.6	Data Volume/Performance	8
2.2.7	Failure Procedure/Operational Instructions.....	8
2.2.8	Programs Impacted	8
2.3	Overpayment Adjustment Logic	8
2.3.1	Overview.....	8
2.3.2	Description of Changes.....	8
2.4	Tax Intercept (CIV Only)	9
2.5	Generate a List for Recovery Accounts	9
3	Supporting Documents	10
4	Requirements.....	10
4.1	Project Requirements	10
4.2	Migration Requirements.....	10

5 Migration Impacts 10
6 Outreach 11
7 Appendix 11

DRAFT

1 OVERVIEW

This SCR describes the policy changes as described in the ACL 19-19 regarding the implementation of SB 726. Effective July 1, 2019, SB 726 will increase the threshold for **non-fraudulent** CalWORKs overpayments from \$35 to \$250.

The changes to the CalWORKs overpayment process pursuant to SB 726 also apply to the Refugee Cash Assistance (RCA) including Trafficking and Crime Victims Assistance Program(TCVAP).

The CalWORKs Overpayment Discharge process for **non-fraudulent** overpayments and the corresponding noticing requirements will be implemented in a later release with CA-207051 CIV-103623.

1.1 Current Design

The CalACES system does not demand the collection of CalWORKs Overpayments totaling less than \$35 from individuals responsible for the recovery account. These recovery accounts where the original claim balance is \$35 or less are marked as Terminated.

1.2 Request

Enhance the CalACES system to increase the threshold for CalWORKs overpayments from \$35 to \$250. ACL 19-19 instructs that the overpayment threshold policy applies only to closed cases and therefore, Effective July 1, 2019 the CalACES system shall not demand collection of any **non-fraudulent** overpayments totaling less than \$250 if the responsible individual(s) is no longer aided in the System under an active CalWORKs or RCA program including TCVAP.

1.3 Overview of Recommendations

This system change request will:

- 1) Introduce a new Recovery Account Investigations value in the system.
- 2) Enhance the Uncollectible Recovery Account Batch to update the **non-fraudulent** recovery accounts for CalWORKs or RCA program including TCVAP with an Outstanding balance less than \$250 as 'Suspended' and status reason as 'Policy Threshold Limit'.
- 3) Update the Overpayment Adjustment Logic to automatically start Benefit Reductions for those **non-fraudulent** Recovery Accounts that have an outstanding balance less than \$250 and the recovery account Status is 'Suspended' with the status reason of 'Policy Threshold Limit'.

1.4 Assumptions

None.

2 RECOMMENDATIONS

2.1 Recovery Account Detail

2.1.1 Overview

This update is to add new Recovery Account Investigations value in the system.

2.1.2 Description of Changes

- 1) Add a new option of 'No Fraud' within the Investigations field on the Recovery Account Detail page. This new option of 'No Fraud' will be selectable or editable for Recovery Accounts in the following Status:
 - a. Pending
 - b. New (CalACES)
 - c. Pending Approval (CalACES)
 - d. Active
 - e. Suspended
 - f. Pending Agreement (CalACES)

NOTE:

1. No Data Change will be involved in the creation of this new Recovery Account Investigations value.
2. Currently, the only field editable on Recovery Account Detail page when the Recovery Account is in Suspended Status is the Comments Section.
- 2) Update the Code Hierarchy between 'Suspended' status and the Status Reason of 'Policy Threshold Limit'.
- 3) End date the existing Status Reason 'Cash - Under \$35 – AE'.

2.1.3 Page Location

No Change.

2.1.4 Counties Impacted

All CalACES Counties.

2.1.5 Security Updates

No Change.

2.1.6 Page Mapping

No Change.

2.2 Uncollectible Recovery Account Batch

2.2.1 Overview

This enhancement is to update the Uncollectible Recovery Account Batch to update the **non-fraudulent** CalWORKs or RCA including TCVAP recovery accounts with an outstanding balance of less than \$250 as 'Suspended' with the status reason of 'Policy Threshold Limit' provided the responsible individual(s) are **no longer receiving aid** in the System under an active CalWORKs or RCA program (including TCVAP).

2.2.2 Description of Changes

- 1) Update the batch to find CalWORKs/RCA Admin Error or Client Error Recovery Accounts with an outstanding balance between \$0 and \$249 that are in 'Active' or 'Suspended' status and status reason is not 'Policy Threshold Limit', where the following conditions are true:
 - a. There is no active CalWORKs/RCA program (including TCVAP) in the System for a Responsible individual of the recovery account.
 - b. The Investigations Indicator is set to 'None' or 'No Fraud'.

Update the Status to 'Suspended' and Status Reason to 'Policy Threshold Limit'.

Following are the cause codes that batch will track:

- a. Cash - Admin Caused
- b. Cash - Customer Caused
- c. Cash - Late QR7
- d. Cash - Late SAR7

- 2) Add a comment on the Recovery Account once the Status is updated to Suspended.

2.2.3 Execution Frequency

No Change.

2.2.4 Key Scheduling Dependencies

No Change.

2.2.5 Counties Impacted

All CalACES Counties.

2.2.6 Data Volume/Performance

No Change.

2.2.7 Failure Procedure/Operational Instructions

No Change.

2.2.8 Programs Impacted

CalWORKs and RCA.

2.3 Overpayment Adjustment Logic

2.3.1 Overview

The overpayment adjustment is a process where if an individual is overpaid and is eligible for benefits, then the system makes an adjustment. This adjustment is a percentage-based calculation per the cause code of the recovery account. This helps in reducing the balance of recovery account. The transaction is posted as a Benefit Reduction.

This Update is to automatically start Benefit Reductions for **non-fraudulent** Recovery Accounts that have an outstanding balance less than \$250 and the recovery account Status is 'Suspended' with the status reason of 'Policy Threshold Limit'.

2.3.2 Description of Changes

- 1) Update the Overpayment Adjustment Logic to automatically start Benefit Reductions for those **non-fraudulent** Recovery Accounts that have an outstanding balance less than \$250 and the recovery account Status is 'Suspended' with the status reason of 'Policy Threshold Limit'.

NOTE: C-IV Counties currently do benefit reductions on 'Suspended' recovery accounts only for the following status reasons, whereas LA County only on 'Active' recovery accounts.

- a. Tax Intercept
- b. Cost
- c. Expired Statute
- d. Social Security

- e. Collection Agency
- f. Unable to Locate
- g. Prison
- h. Admin Decision

With this change request (CA 205172 CIV 102502) there would be no impact to the current C-IV Counties functionality and they would add 'Policy Threshold Limit' status reason to the existing list of Status Reason for Suspended status. However, LRS will now start doing Benefit Reductions on Suspended recovery accounts with Status Reason of 'Policy Threshold Limit' Only.

NOTE: This is an already identified GAP between the two systems that will be addressed as part of a different SCR where all the Region would decide collectively to determine what changes will be migrated into CalSAWS.

2.4 Tax Intercept (CIV Only)

Update the logic for Tax Intercept job to send a delete record for recovery accounts that are currently in Tax Intercept and will be Suspended due to 'Policy Threshold Limit' because of new policy.

2.5 Generate a List for Recovery Accounts

A list of recovery accounts will be generated as part of this change request and will be provided before the change is implemented in the production. The list will identify the impacted records based on following criteria:

1. Recovery Account status is 'Active' or 'Suspended' and status reason is not 'Policy Threshold Limit'.
2. Recovery Account is associated the CalWORKs or RCA program.
3. The Cause code is either Admin error or Client error.
4. The Outstanding Balance is totaling less than \$250.
5. The Outstanding Balance is greater than \$0.
6. The Investigations Indicator is 'None'.
7. The responsible individual(s) is not aided in California under the CalWORKs or RCA program.

The listing will be provided to the counties for their review. When this SCR is implemented in production the Uncollectible recovery account batch will automatically update all the recovery accounts identified in the list as 'Suspended' and Status Reason as 'Policy Threshold Limit' because of the new policy implementation.

NOTE: This listing will cover both System created as well as External recovery accounts falling into the criteria specified above.

3 SUPPORTING DOCUMENTS

Number	Functional Area	Description	Attachment

4 REQUIREMENTS

4.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
2.11.2.12	The LRS shall discontinue collection of overpayments/over issuances once all outstanding overpayments/overissuances have been collected, there is an overpayment/overissuance claim status change to terminated, voided, or uncollectable, or the collection of the overpayment/overissuance has been deemed satisfied.	This Requirement is met by enhancing the Uncollectible Recovery Account Batch to update the CalWORKs non-fraudulent recovery accounts with an Outstanding balance under \$250 as 'Suspended' with the status reason of 'Policy Threshold Limit'.

4.2 Migration Requirements

DDID #	REQUIREMENT TEXT	How Requirement Met
N/A		

5 MIGRATION IMPACTS

Number	Functional Area	Description	Impact	Priority	Address Prior to Migration?
N/A					

6 OUTREACH

Provide Counties with a list of recovery accounts falling in below criteria:

1. The Recovery Account status is either 'Active', 'Suspended'.
2. Recovery Account was established on or after December 1, 1996
3. Recovery Account is associated the CalWORKs or RCA program.
4. The Cause code is either Admin error or Client error.

5. The Investigations Indicator is 'Investigations'.
6. The responsible individual(s) have not been aided in the System under the CalWORKs or RCA program in the past consecutive 36 months.

7 APPENDIX


None.

DRAFT



Design Document

SCR 102054 – C4Y: Update Sending Documents
to include County for Mobile App and
C4Yourself

	DOCUMENT APPROVAL HISTORY	
	Prepared By	Gerald Limbrick
	Reviewed By	[individual(s) from build and test teams that reviewed document]

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
4/5/19	1.0	Initial Document	Gerald Limbrick

DRAFT

Table of Contents

1	Overview	4
1.1	Current Design.....	4
1.2	Requests	4
1.3	Overview of Recommendations	4
1.4	Assumptions	5
2	Recommendations	6
2.1	C4Yourself	6
2.1.1	Overview.....	6
2.1.2	C4Yourself Mockups.....	7
2.1.3	Description of Changes.....	10
2.1.4	Page Location	11
2.2	Mobile App – Upload Files	12
2.2.1	Overview.....	12
2.2.2	Mobile App - Upload Files Mockups	13
2.2.3	Description of Changes.....	15
2.2.4	Page Location	16
3	Supporting Documents	17
4	Requirements.....	18
4.1	Project Requirements	18
4.2	Migration Requirements.....	18
5	Migration Impacts.....	19
6	Appendix.....	20

1 OVERVIEW

This SCR will reduce the number of documents associated to the wrong county when uploading through the self-service portal.

1.1 Current Design

When uploading a document through C4Yourself or the C4Yourself Mobile App, the customer first chooses to link the document by case or e-app. They then choose from all cases or submitted non "RE" e-applications associated to their login. This includes old cases from previous counties. Many customers are incorrectly choosing to link their document to an older case, associating it to the wrong county.

Under some conditions the required field indicator "*" looks out of alignment with other labels on the page.

A New Uploaded Document(s) message is sent to the C4Yourself Message Inbox after a customer uploads a document. This message includes the document type(s), but it does not include the county or case associated to the document(s).

1.2 Requests

Have the customer select which county a document should be sent to, before choosing whether to link the document by case or e-application. After choosing a county, only allow the customer to choose from the case numbers or e-application numbers for the chosen county.

When the customer chooses to link document by case, display the programs associated with that case.

Display a verification pop-up if the customer is uploading to a case which does not have either an active or pending program or a program discontinued in the last 90 days.

Include the county and case number chosen in the New Uploaded Document(s) message, sent after an upload.

1.3 Overview of Recommendations

- Update C4Yourself and the C4Yourself Mobile App to include a county drop down, filtered by counties in which they have associated cases or e-applications.
- In both applications, on the case drop downs, show an indicator if they have at least one program active or pending on that case.
- In both applications, display the case numbers or e-applications in the respective dropdowns dynamically based on the county selected.
- In both applications when the customer chooses to link a document by case, display the programs associated to that case; this will help the customer decide if they chose the right case/county.
- In both applications, display a verification pop-up if the customer is uploading to a case which does not have either an active or pending program or a program discontinued in the last 90 days.

- Add the county and case number chosen to the New Uploaded Document(s) message.

1.4 Assumptions

None.

DRAFT

2 RECOMMENDATIONS

2.1 C4Yourself

2.1.1 Overview

Currently, when uploading a document, the customer first chooses to link the document by case or e-app. They then choose from all cases or submitted non "RE" e-applications associated to their login. The customer can then select from any of the case numbers associated to them (linked to their login). This includes old case numbers from previous counties. Many customers are incorrectly choosing to link their document to an older case, associating it to the wrong county.

There is a space between the required field indicator "*" and the Case Number field label; other required fields on the page do not have such a space.

After choosing to link the document to either a case or e-application, the required field indicator displays after the field label thus it looks out of alignment with other labels on the page.

To reduce the number of documents associated to the wrong county, add a dropdown with counties associated (thru case or e-application) with the customer's login. Populate the Case Number dropdown or E-App Number dropdown dynamically based on the county selected. Display "Active", "Pending" or "Active/Pending" after case numbers in the dropdown as applicable. When the customer chooses to link document by case, display an additional row with a list of programs associated to the case, to help the customer decide if they chose the right case/county.

Display a verification pop-up if the customer is uploading to a case which does not have either an active or pending program or a program discontinued in the last 90 days.

Make cosmetic updates to consistently display the required field "*" indicator.

2.1.2 C4Yourself Mockups

Please provide information so that we can link your document to the appropriate case.

Note: use of this site is limited solely to legitimate C4Yourself® purposes and any documents or files submitted electronically will become part of the user's case file. Submission of any inappropriate material is strictly prohibited. Inappropriate material includes, but is not limited to, the following: viruses or malicious code; content that is commercial, pornographic, contains nudity, threatening, violent, discriminatory or unlawful; copyrighted material used without permission of the owner; content or code which attempts to gather the personal information of other users ("phishing"). Submission of unlawful material will be referred to the appropriate authorities.

Document Information

Send Documents

The screenshot shows a web form titled "Send Documents" with the following fields and a dropdown menu:

- *County: San Bernardino (dropdown)
- *Link Document To: Case (dropdown)
- *Case Number: (dropdown menu is open, showing options: Select One, 2037439 Active, 1574284 Active/Pending, 2649478 Pending, 2185112)
- Program(s): (text input)
- *Document Type: (dropdown)
- First Name: First (text input)
- Last Name: Last (text input)
- *Document: (text input) Browse... (button)

Continue

Figure 2.1.1 – Send Documents – Case Dropdown Mockup

Please provide information so that we can link your document to the appropriate case.

Note: use of this site is limited solely to legitimate C4Yourself® purposes and any documents or files submitted electronically will become part of the user's case file. Submission of any inappropriate material is strictly prohibited. Inappropriate material includes, but is not limited to, the following: viruses or malicious code; content that is commercial, pornographic, contains nudity, threatening, violent, discriminatory or unlawful; copyrighted material used without permission of the owner; content or code which attempts to gather the personal information of other users ("phishing"). Submission of unlawful material will be referred to the appropriate authorities.

Document Information

Send Documents

*County:	San Bernardino	▼
*Link Document To:	Case	▼
*Case Number:	1574284 Active/Pending	▼
Program(s):	CalFresh, CalWorks, Medical	
*Document Type:	Select One	▼
First Name:	First	
Last Name:	Last	
*Document:		Browse...

Continue

Figure 2.1.2 – Send Documents – Program(s) List Mockup

Please provide information so that we can link your document to the appropriate case.

Note: use of this site is limited solely to legitimate C4Yourself® purposes and any documents or files submitted electronically will become part of the user's case file. Submission of any inappropriate material is strictly prohibited. Inappropriate material includes, but is not limited to, the following: viruses or malicious code; content that is commercial, pornographic, contains nudity, threatening, violent, discriminatory or unlawful; copyrighted material used without permission of the owner; content or code which attempts to gather the personal information of other users ("phishing"). Submission of unlawful material will be referred to the appropriate authorities.

Document Information

Send Documents

*County:

*Link Document To:

*Case Number:

Program(s):

*Document Type: Select One

First Name: First

Last Name: Last

*Document: Browse...

Continue

There are
NO ACTIVE PROGRAMS for
this case in the last 90 days.

Continue Anyway?

Cancel Continue

Figure 2.1.3 – Verification Pop-Up Mockup

Message Inbox

Subject	Date	
<u>New Uploaded Document(s)</u>	01/21/2019	Remove
The following document types have been uploaded to San Bernardino for Case Number 123456 :		
1) Other ID Cards		Close
2) Social Security Card		
<u>New Uploaded Document(s)</u>	11/08/2018	Remove
<u>New Uploaded Document(s)</u>	11/08/2018	Remove
<u>Your link request was unable to be completed.</u>	10/24/2018	Remove
<u>Thank you for submitting your registration request</u>	10/23/2018	Remove

Figure 2.1.4 – Message Inbox Mockup

2.1.3 Description of Changes

- 1) Add a county dropdown as the first row of the Document Information form.
 - a) Label the row "County:"
 - b) Mark this row with the required field indicator "*".
 - c) Populate this dropdown with a filtered list of counties, showing only counties having at least one case or e-application linked to the customer's login.
 - d) Allow only one county to be selected per document upload.
 - e) Default the county dropdown to an empty "Select One" value, requiring the customer to choose a county.
 - f) If the customer does not choose a county, display the validation message: "You need to select a value from this list."
- 2) Update the Case Number and E-App Number dropdowns to display a dynamically filtered list based on the county selected.
- 3) On the Case Number drop down:
 - a) Add the word "Active" after any case number that has at least one active program.
 - b) Add the word "Pending" after any case number that has at least one pending program.
 - c) If a case has both active and pending programs, display "Active/Pending" after the case number.
 - d) Sort the list of cases:
 - i) Show cases with at least one active program at the top of the list.
 - ii) Next, show cases having at least one pending program but none active.
 - iii) Put any cases with neither active nor pending programs at the bottom of the list.

- 4) Add a non-editable program row below the Case Number dropdown.
 - a) Label the row "Program(s):"
 - b) Display this row only when the customer chooses to link the document to a case.
 - c) Display a comma separated list of programs associated to the chosen case.
 - d) Display the list of programs as text, "CalFresh", "CalWORKs" or "Medical" as applicable.
- 5) Display a verification pop-up if the customer chose to upload the document to a case having neither an active or pending program nor a program discontinued in the past 90 days. Note: This is not applicable when linking by e-application because the customer can only choose submitted e-applications.
 - a) Display the pop-up on clicking Continue.
 - b) In the pop-up, display the text "There are NO ACTIVE PROGRAMS for this case in the last 90 days. Continue Anyway?" followed by buttons "Cancel" and "Continue".
 - i) Capitalize and embolden the words "NO ACTIVE PROGRAMS".
 - ii) If the customer clicks Cancel they will stay on the current page having the opportunity to change their selections.
 - iii) If the customer clicks Continue, they will continue in the upload document flow as before. Note: The customer CAN upload to an older case.
- 6) Remove the space between the required field indicator "*" and the Case Number field label.
- 7) Update the way the required field indicator displays after choosing from the Link Document To dropdown; display the required field indicator before, instead of after, the Case Number or E-App Number field labels.
- 8) Update the New Uploaded Document(s) message to include the County and Case number chosen. The new message body will read either:
 - a) "The following document types have been uploaded to <County_Name> for **Case Number <Case_Number>**:
<document_Type_List>"
 - Or
 - b) "The following document types have been uploaded to <County_Name> for **Case Number <Case_Number>** through Mobile App:
<document_Type_List>"
 - c) Embolden the county name and "Case Number <case_number>".
- 9) Update all threshold languages.

2.1.4 Page Location

C4Yourself > My Documents

2.2 Mobile App – Upload Files

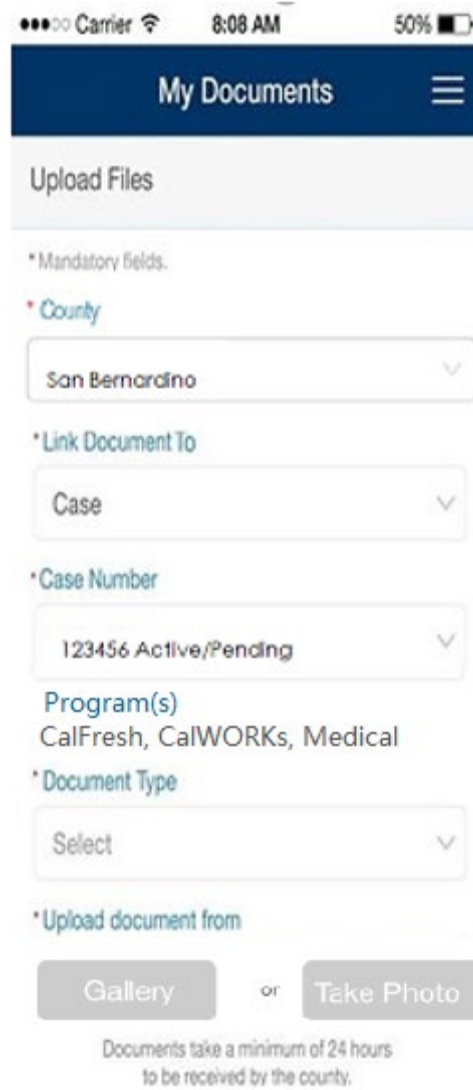
2.2.1 Overview

Currently, when uploading a document, the customer first chooses to link the document by case or e-app. They then choose from all cases or submitted non “RE” e-applications associated to their login. The customer can then select from any of the case numbers associated to them (linked to their login). This includes old case numbers from previous counties. Many customers are incorrectly choosing to link their document to an older case, associating it to the wrong county.

To reduce the number of documents associated to the wrong county, add a dropdown with counties associated (thru case or e-application) with the customer’s login. Populate the Case Number dropdown or E-App Number dropdown dynamically based on the county selected. Display “Active”, “Pending” or “Active/Pending” after case numbers in the dropdown as applicable. When the customer chooses to link document by case, display an additional field with a list of programs associated to the case, to help the customer decide if they chose the right case/county.

Display a verification pop-up if the customer is uploading to a case which does not have either an active or pending program or a program discontinued in the last 90 days.

2.2.2 Mobile App - Upload Files Mockups



Carrier 8:08 AM 50%

My Documents

Upload Files

* Mandatory fields.

* **County**

San Bernardino

* **Link Document To**

Case

* **Case Number**

123456 Active/Pending

Program(s)
CalFresh, CalWORKs, Medical

* **Document Type**

Select

* **Upload document from**

Gallery or Take Photo

Documents take a minimum of 24 hours to be received by the county.

Figure 2.2.1 – Mobile App - Upload Files Mockup

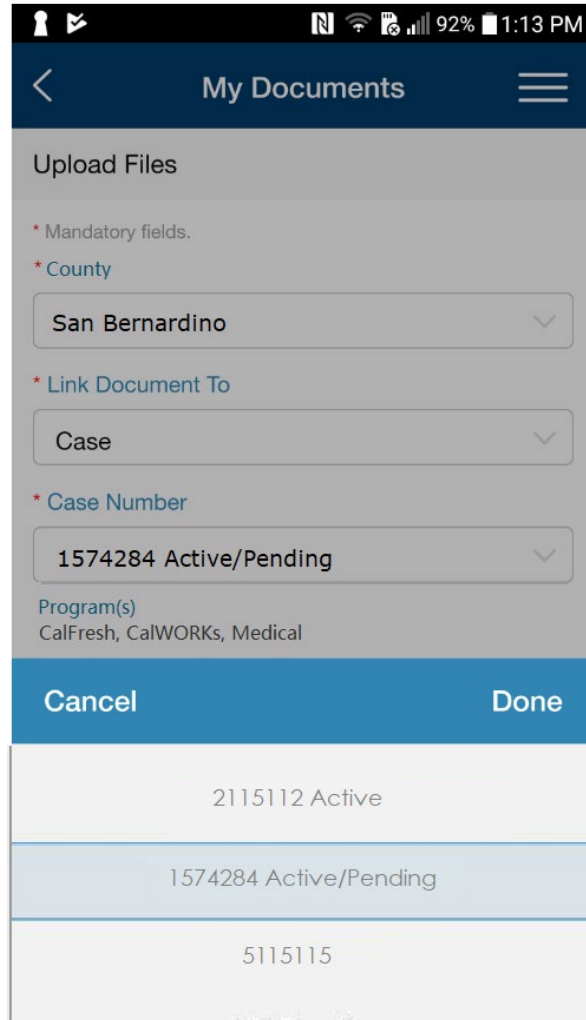


Figure 2.2.2 – Mobile App – Case Dropdown Mockup

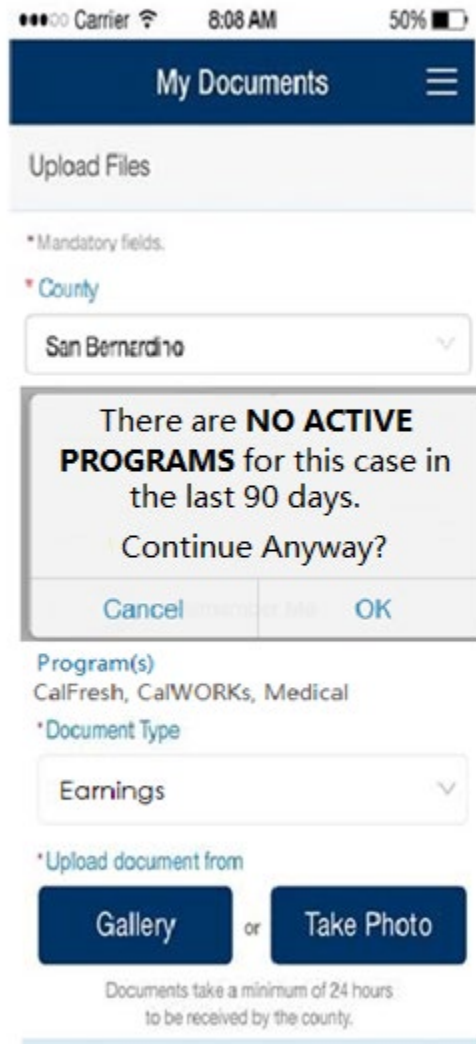


Figure 2.2.3 – Verification Pop-Up Mockup

2.2.3 Description of Changes

- 1) Add a county dropdown and label at the top of the Upload Files screen.
 - a) Label the dropdown "County".
 - b) Mark the label with the required field indicator "*".
 - c) Allow only one county to be selected per document upload.
 - d) Default the county dropdown to an empty "Select" value, requiring the customer to choose a county.
 - e) Populate this dropdown with a filtered list of counties; show only counties having a case or e-application linked to the customer's login.
- 2) Update the Case Number and E-Application Number dropdowns to display a dynamically filtered list based on the county selected.
- 3) On the Case drop down:

- a) Add the word "Active" after any case number that has at least one active program.
 - b) Add the word "Pending" after any case number that has at least one pending program.
 - c) If a case has both active and pending programs display "Active/Pending" after the case number.
 - d) Sort the list of cases:
 - i) Show cases with at least one active program at the top of the list.
 - ii) Next, show cases having at least one pending program but none active.
 - iii) Put any cases with neither active nor pending programs at the bottom of the list.
- 4) Add a non-editable Program field below the Case Number dropdown.
 - a) Label the row "Program(s)".
 - b) Display this row only when the customer chooses to link the document to a case.
 - c) Display a comma separated list of programs associated to the chosen case.
 - d) Display the list of programs as text: "CalFresh", "CalWORKs" or "Medical" as applicable.
 - 5) Display a verification pop-up if the customer chose to upload the document to a case having neither an active or pending program nor a program discontinued in the past 90 days. Note: This is not applicable when linking by e-application because the customer can only choose submitted e-applications.
 - a) Display the pop-up on clicking Gallery or clicking Take Photo.
 - b) In the pop-up, display the text "There are NO ACTIVE PROGRAMS for this case in the last 90 days. Continue Anyway?" followed by buttons "Cancel" and "Continue".
 - i) Capitalize and embolden the words "NO ACTIVE PROGRAMS".
 - ii) If the customer clicks Cancel they will stay on the current page having the opportunity to change their selections.
 - c) If the customer clicks OK, they will continue in the upload document flow as before. Note: The customer CAN upload to an older case.
 - 6) Update the New Uploaded Document(s) message to include the County and Case number chosen. The new message body will read either:
 - a) "The following document types have been uploaded to **<County_Name>** for **Case Number <Case_Number>**:
<document_Type_List>"
Or
 - b) "The following document types have been uploaded to **<County_Name>** for **Case Number <Case_Number>** through Mobile App:
<document_Type_List>"
 - c) Embolden the county name and "Case Number <case_number>".
 - 7) Update Spanish pages.

2.2.4 Page Location

C4Yourself > My Documents

3 SUPPORTING DOCUMENTS

[This section should include any supporting documents for the design as imbedded documents. Some examples of supporting documents include the Security Matrix, Form Design Documents, NOA Design Documents, and ETL Source-to-Target Mappings.]

Number	Functional Area	Description	Attachment

DRAFT

4 REQUIREMENTS

[Document what requirements are being addressed with this design and how they are being met]

4.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met

4.2 Migration Requirements

DDID #	REQUIREMENT TEXT	How Requirement Met

DRAFT

5 MIGRATION IMPACTS

None.

SCR Number	Functional Area	Description	Impact	Priority	Address Prior to Migration?

DRAFT

6 APPENDIX

[Include any supplementary items that may not fit in the Description section. Examples could include flow charts, lengthy code tables, etc....]

DRAFT

CalSAWS

California Statewide Automated Welfare System

Design Document

SCR 102444 – CAAL-Skills Pilot Data Exchange

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Duke Vang
	Reviewed By	Chris Carandang

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
4/4/2019	1.0	Initial Revision	Duke Vang
4/23/2019	1.1	Added additional information and fixed typos after review from Chris Carandang	Duke Vang

DRAFT

Table of Contents

1	Overview	4
1.1	Current Design.....	4
1.2	Requests	4
1.3	Overview of Recommendations	4
1.4	Assumptions	5
2	Recommendations	6
2.1	CAAL-Skills Data Extract Specifications	6
2.1.1	Global Definitions and Variables.....	6
2.1.2	Description of Change	7
2.1.3	Counties Impacted	18
3	Supporting Documents	19
4	Appendix.....	20

DRAFT

1 OVERVIEW

The California Workforce Development Board (CWDB) has created the CAAL-Skills project to develop a comprehensive, integrated, interoperable data system that measures and assesses program outcomes in a systematic and efficient manner. Per the Memorandum of Understanding (MOU) between the California Department of Social Services (CDSS) and CalSAWS (CalSAWS, C-IV, and LRS), the confidential CalWORKs (CW) and CalFresh (CF) client data of Napa, Monterey, and Stanislaus County will be provided to CDSS to be utilized by the CAAL-Skills project in a pilot run.

1.1 Current Design

The C-IV System does not exchange any information with the CAAL-Skills project.

1.2 Requests

Per the CalSAWS and CAAL-Skills MOU Agreement (see Supporting Documentation 1), all CW and CF data for Napa, Monterey, and Stanislaus County from 7/1/2013 to 6/30/2017 (report period) will be transmitted to CDSS based on the data specified under the CAAL-Skills Data Exchange Requirements (see Supporting Documentation 2)

Note: In working with CWDB on the data specifications from the CAAL-Skills Data Exchange Requirements, there have been agreed upon deviations to the data specifications between CalSAWS and CWDB (see Supporting Document 5). These deviations will be called out and covered in detail in each recommendation below.

1.3 Overview of Recommendations

The client confidential data of all CW and CF for Napa, Monterey, and Stanislaus County will be sent in five separate Comma-Separated Value (CSV) files. Each of the records from each file will be linkable to records from other files with the use of imbedded "foreign keys" within each file. The five files and their high-level contents are:

1. Admin Agency/Partner
 - a. Admin Agency ID
2. Enrolling Organization
 - a. Enrolling Organization Name
 - b. Enrolling Organization ID
3. Services
 - a. Funding Source
 - b. Career Services
 - c. Training Services
 - d. Employment & Training Services
4. Supportive Services
 - a. Type of Supportive Services
 - b. First Day of Supportive Services
 - c. Last Day of Supportive Services

5. Participant
 - a. Social Security Number (SSN)
 - b. First Name
 - c. Middle Name
 - d. Last Name
 - e. Date of Birth
 - f. City
 - g. Zip Code
 - h. State
 - i. Sex
 - j. Client Index Number (CIN)
 - k. Ethnicity
 - l. Race
 - m. Veteran Status
 - n. Homeless Status
 - o. Former Foster Care Youth
 - p. Disability Status
 - q. Migrant Farmer
 - r. Single Parent Status

1.4 Assumptions

This is a onetime data exchange with CDSS. Any future data exchanges will need to go through another MOU agreement between CalSAWS and CDSS.

2 RECOMMENDATIONS

2.1 CAAL-Skills Data Extract Specifications

2.1.1 Global Definitions and Variables

For the remainder of this document, the following definitions and variables will apply to all the data fields and data files (see Supporting Document 7 for mock file examples):

1. Data Type: INT – Numeric value
 - a. As Binary Value
 - i. 0 – Negative
 - ii. 1 – Positive
 - b. As Actual Source Value
 - i. 97 – Other
 - ii. 98 – Not Applicable
 - iii. 99 – Unknown
2. Data Type: VCHAR(NN) – Variable length alpha-numeric data up to NN length.
 - a. All alpha characters will be in all CAPS
 - b. All characters will be truncated after NN length
 - c. As Actual Source Value
 - i. 97 – Other
 - ii. 98 – Not Applicable
 - iii. 99 – Unknown
3. Data Type: DATE – Date Element
 - a. All DATE format will follow MMDDYYYY with left padded 0's for single digit months and days (i.e. 01012019).
4. Data Fields – All data fields will have no padding and will not be right or left justified. Any characters that exceeds the data fields length will be truncated.
5. Null Values – Null values will be represented as blank (i.e. not white space or empty quotes).
6. First Row – The first row of any file will always be the column header.
7. Delimiters – All column header (first row) and data fields will be delimited by a pipe '|'. Null fields will still be delimited by a pipe ('|'). The last character of ANY row will be a pipe including the column header (first row). Any actual source values containing a pipe character ('|') must be surrounded in double quotes ("|") to not confuse the actual value with the pipe delimiter.
8. Last Row – The last row of any file will not contain a carriage return.
9. Filename: CS1_MMDDYYYY_XXXX_NNN_RNN.CSV
 - a. MMDDYYYY – This will be the end date of the file extract. It will always be 06302017.
 - b. XXXX – This will be the name of the file type.
 - i. AAPA – Admin Agency/Partner

- ii. ENRO – Enrolling Organization
 - iii. PART – Participant
 - iv. SERV – Services (Training and Education)
 - v. SUPP – Supportive Services
 - c. NNN – The sequential numbering of the file number.
 - d. RNN – The request number. This will always be R01.
10. File Size – Each of the five data files are not to exceed 250 megabytes (MB). If a file exceeds 250 MB, the file will need to be broken into two or more files and have its filename incremented with the appropriate number (NNN).

For example: CS1_MMDDYYYY_XXXX_ **NNN** _RNN.CSV
MMDDYYYY where NNN is incremented for each subsequent file of the same type.

2.1.2 Description of Change

1. Create a CSV file for the Admin Agency/Partner File. The Admin Agency/Partner file will contain **exactly** three rows of the following element (one for each of the pilot counties):
 - a. Filename: CS1_06302017_AAPA_001_R01.CSV
 - i. **AAID:** (DED #000 of CAAL-Skills Data Exchange Requirements)
 1. Data Type: INT
 2. Name: Admin Agency ID
 3. Values:
 - 10 – CWDA Napa County
 - 11 – CWDA Monterey County
 - 12 – CWDA Stanislaus County

Example:

```
AAID |
10 |
11 |
12 |
```

2. Create a CSV file for the Enrolling Organization file. The Enrolling Organization file will contain **exactly** three rows of the following elements (one for each of the pilot counties):
 - a. Filename: CS1_06302017_ENRO_001_R01.CSV
 - i. **Fieldname: AAID:** (DED #000 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Document 3). This element was added in as a foreign key after talking with CWDB to link records from the ENRO file to the AAPA file)**
 1. Data Type: INT

2. Name: Admin Agency ID
3. Values:
 - 10 – CWDA Napa County
 - 11 – CWDA Monterey County
 - 12 – CWDA Stanislaus County
- ii. **Fieldname: EONM:** (DED #002 of CAAL-Skills Data Exchange Requirements)
 1. Data Type: VCHAR(40)
 2. Name: Enrolling Organization Name
 3. Values:
 - NAPA
 - MONTEREY
 - STANISLAUS
- iii. **Fieldname: EOID:** (DED #003 of CAAL-Skills Data Exchange Requirements)
 1. Data Type: VARCHAR(40)
 2. Name: Enrolling Organization ID
 3. Values:
 - 28 – Napa
 - 27 – Monterey
 - 50 - Stanislaus

Example:
 AAID | EONM | EOID |
 10 | NAPA | 28 |
 11 | MONTEREY | 27 |
 12 | STANILAUS | 50 |

3. Create a CSV file for the Participant file. The population for this file will be any Persons that was an Active Member in a CW or CF (including Transitional CalFresh) program during the report period and the CW or CF program belonged to a Napa, Monterey, or Stanislaus County Case (Refer to Appendix 1 for a technical POC). The Persons from this population will be unduplicated if the Person came on and off aid multiple times during the report period. Furthermore, if the Person made an Inter-County Transfer (ICT) between Napa, Monterey, and Stanislaus County, only the most recent county that the Person was Active during the report period will be pulled into the population. The Participant file will contain the following demographics information for each of the Persons identified in the population as defined in the CAAL-Skills Data Exchange Requirements:

- a. Filename: CS1_06302017_PART_001_R01.CSV

- i. **Fieldname: EOID:** (DED #003 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Document 3). This element was added in as a foreign key after talking with CWDB to link records from the PART file to the ENRO file)**

1. Data Type: VARCHAR(40)
 2. Name: Enrolling Organization ID
 3. Value: This will be the County Code value of the County the last time the Person was a CW or CF Active Member during the report period.
- ii. **Fieldname: PUID** (DED #015 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(20)
 2. Name: Participant Unique ID
 3. Value: This will be the Person's CIN. If the Person does not have a CIN, a '99' will be populated instead. (Refer to Appendix 9 for a technical POC)
- iii. **Fieldname: SSN** (DED #005 of CAAL-Skills Data Exchange Requirements)
1. Data Type: INT
 2. Name: SSN
 3. Value: This will be the Person's SSN. If the Person does not have a SSN, a '99' will be populated instead. (Refer to Appendix 9 for a technical POC)
- iv. **Fieldname: PFNM** (DED #006 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(40)
 2. Name: Participant First Name
 3. Value: This will be the Person's First Name. (Refer to Appendix 9 for a technical POC)
- v. **Fieldname: PMNM** (DED #007 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(40)
 2. Name: Participant Middle Name
 3. Value: This will be the Person's Middle Name. (Refer to Appendix 9 for a technical POC)
- vi. **Fieldname: PLNM** (DED #008 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(40)
 2. Name: Participant Last Name
 3. Value: This will be the Person's Last Name. (Refer to Appendix 9 for a technical POC)
- vii. **Fieldname: DOB** (DED #013 of CAAL-Skills Data Exchange Requirements)
1. Data Type: DATE
 2. Name: Date of Birth
 3. Value: This will be the Person's Date of Birth. (Refer to Appendix 9 for a technical POC)
- viii. **Fieldname: SEX** (DED #014 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(2)

2. Name: Participant's Sex
 3. Value: This will be the Person's biological sex. A Male value will be 'M' and a Female value will be 'F'. If the gender is unknown or is null, the value will be a blank. (Refer to Appendix 9 for a technical POC)
- ix. **Fieldname: PCITY** (DED #010 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(50)
 2. Name: Participant City Address
 3. Value: This will be the last City that the Person resided during the report period. If the person has a different mailing and physical address, the hierarchy will be: Physical -> Mailing. (Refer to Appendix 2 for a technical POC)
- x. **Fieldname: PPOST** (DED #011 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(9)
 2. Name: Participant Postal Zip Code
 3. Value: This will be the last Postal Zip Code that the Person resided during the report period. The Postal Zip Code will include the zip code plus the 4-digit suffix minus any special characters. For example: 85602-1023 will become 8356021023. If the person has a different mailing and physical address, the hierarchy will be: Physical -> Mailing. (Refer to Appendix 2 for a technical POC)
- xi. **Fieldname: PST** (DED #012 of CAAL-Skills Data Exchange Requirements)
1. Data Type: VCHAR(2)
 2. Name: Participant State Address
 3. Value: This will be the last State that the Person resided during the report period. The value will be the abbreviation of the State. If the person has a different mailing and physical address, the hierarchy will be: Physical -> Mailing. (Refer to Appendix 2 for a technical POC)
- xii. **Fieldname: VET** (DED #018 of CAAL-Skills Data Exchange Requirements)
1. Data Type: INT
 2. Name: Participant Veteran Status
 3. Value: This will be the Person's Veteran Status. If the Person has a Military Service record within the System, the Person will be considered a Veteran and the value will be set to '1'. If the Person does not have a Military Service record within the System, the value will be set to '0'. (Refer to Appendix 9 for a technical POC)

- xiii. **Fieldname: ETHN** (DED #017 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement. Per meeting with CWDB from 3/22/2019 (See Supporting Document 4), Race vs. Ethnicity is not clearly defined at the State level. This data element will contain both Race and Ethnicity values. CWDB will work with CDSS to separate Race from Ethnicity later once the data has been collected)**
1. Data Type: VCHAR(100)
 2. Name: Participant Ethnicity
 3. Value: This will be the actual decoded value of the Person's Race and Ethnicity. If the Person identifies as more than one Race or Ethnicity, the actual decoded values will be listed separated by a comma. (Refer to Appendix 3 for a technical POC)
- xiv. **Fieldname: RACE** (DED #017-1 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement. Per meeting with CWDB from 3/22/2019 (see Supporting Document 4), the CalSAWS Project and CWDB agreed to always populate this field with a blank.**)
1. Data Type: VCHAR(100)
 2. Name: Participant Race
 3. Value: This value will always be blank. (Refer to Appendix 3 for a technical POC)
- xv. **Fieldname: HMLS** (DED #020 of CAAL-Skills Data Exchange Requirements)
1. Data Type: INT
 2. Name: Homeless Individual
 3. Value: This will be the Person's Homeless Status. If the Person has a Living Arrangement of either "Homeless" or "Homeless Shelter" any time during the report period, the value will be set to '1'. If the person has no Living Arrangement data point, the value will be set to '0'. All other Living Arrangement values will be set to '99'. (Refer to Appendix 4 & 9 for a technical POC)
- xvi. **Fieldname: FCTRC** (DED #023 of CAAL-Skills Data Exchange Requirements)
1. Data Type: INT
 2. Name: Participant Foster Care Youth
 3. Value: This data point will indicate if the Person is a current or former Foster Care Youth. If the Person was ever in a Foster Care program within the System at any point in time, regardless of County, the value will be set to a '1'. Else the value will be

set to a '0'. (Refer to Appendix 6 for a technical POC)

xvii. **Fieldname: DISA** (DED #027 of CAAL-Skills Data Exchange Requirements)

1. Data Type: INT
2. Name: Participant Permanent/Temporary Disability
3. Value: This will be the Person's Disability Status. The value will be set to one of the following values based on the Person's Medical Conditions. If there are more than one Medical Condition, only ONE value will be used to indicate the Person's Disability Status. The hierarchy will be Permanent → Temporary → N/A → Unknown. (Refer to Appendix 5 for a technical POC):

- '0' – No Disability
 - No medical conditions listed
- '1' – Permanent Disability
 - Other CF Disability
 - SP-DDSD Blind
 - SP-DDSD Disabled
 - SP-DDSD Presumptive Approval
 - SSA Blind
 - SSA Disabled
 - SSI Blind
- '2' – Temporary Disability
 - Fry v. Saenz Disability
 - Incapacity - Verifiable and at Least 30 Days
 - Limited Services - Kidney Dialysis
 - Limited Services – TPN
 - Limited Services – Tuberculosis
 - Obviously Unfit for Employment for CalFresh
 - Presumptive Allegation of ALS (Lou Gehrigs)
 - Presumptive Allegation of Bed Confinement
 - Presumptive Allegation of Cerebral Palsy
 - Presumptive Allegation of Down Syndrome
 - Presumptive Allegation of Mental Retardation
 - Presumptive Allegation of Muscular Dystrophy
 - Presumptive Allegation of Stroke

- Presumptive Allegation of Total Blindness
- Presumptive Allegation of Total Deafness
- Presumptive Amputation of a Leg at the Hip
- Presumptive Birth Weight Below 2 lbs.10 oz.
- Presumptive End Stage Renal Disease
- Presumptive Gestational Age/Weight at Birth
- Presumptive Human Immunodeficiency Virus (HIV)
- Presumptive Nonambulatory/Spinal Cord Injury
- Presumptive Terminally Ill
- Unfit for Employment for CalFresh
- '98' – N/A
 - SP-DDSD Pending
- '99' – Unknown
 - FC/KG – Physical or Mental Disability

xviii. **Fieldname: FARM** (DED #028 of CAAL-Skills Data Exchange Requirements)

1. Data Type: INT
2. Name: Participant Migrant Farm Worker
3. Value: This will be the Person's Migrant Farm Worker indicator. If the Person indicated that they are a Migrant Seasonal Farm Worker on the Residency Detail page, then this value will be set to '1', else the value will be set to a '0'. (Refer to Appendix 7 for a technical POC)

xix. **Fieldname: SPAR** (DED #029 of CAAL-Skills Data Exchange Requirements)

1. Data Type: INT
2. Name: Participant Single Parent
3. Value: This will be the Person's Single parent indicator. This value will be determined by using the Person's last CW or CF Aid Code during the report period. If the Person had both a CW and CF Aid Code during the report period, the hierarchy will be: CW Aid Code → CF Aid Code. The value will be set based on the Aid Code as follows. (Refer to Appendix 8 for a technical POC and Supporting Documents 6):
 - '0' – Not a Single Parent
 - i. 33
 - ii. 35

- iii. 3C
- iv. 3F
- v. 3G
- vi. 3H
- vii. 3M
- viii. 3U
- ix. 4N
- x. 4R
- xi. R5
- xii. R8
- '99' – Unknown
 - i. 09
 - ii. 0F
 - iii. 30
 - iv. 32
 - v. 3A
 - vi. 3E
 - vii. 3L
 - viii. 3W
 - ix. 4P
 - x. K1
 - xi. R4
 - xii. R6
 - xiii. R7
 - xiv. R9

4. Create a CSV file for the Service file. The population for this file will be all the Customer Activities that the Persons from the Participant file participated in during the report period.

a. Filename: CS1_06302017_SERV_001_R01.CSV

i. **Fieldname: PUID** (DED #015 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 3). This element was added in as a foreign key after talking with CWDB to link records from the SERV file to the PART file.)**

1. Data Type: VCHAR(20)
2. Name: Participant Unique ID
3. Value: This will be the Person's CIN. If the Person does not have a CIN, a '99' will be populated instead. (Refer to Appendix 9 for a technical POC)

ii. **Fieldname: FSRC** (DED #004 of CAAL-Skills Data Exchange Requirements.)

1. Data Type: INT
2. Name: Funding Source
3. Value: This will be the Funding Source of the Customer Activity. If the Customer Activity is

linked to a CW or Welfare to Work (WT) program, then the value will be set to '7' – CalWORKs. If the Customer Activity is linked to a CF or CalFresh Employment & Training (CFET) program, then the value will be set to '8' – CalFresh. (Refer to Appendix 10 for a technical POC)

- iii. **Fieldname: TCS** (DED #035 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 4). Per meeting with CWDB on 3/22/2018, CWDB did not want CalSAWS to map the Participant Customer Activities into one of their Career Services values. Instead, the Customer Activity data will be provided as “actual values” to CAAL-Skills under DED #035 and CWDB will work with CDSS to map the Customer Activities. In light of these changes, the Data Type has changed from an INT to a VCHAR(40).**)

1. Data Type: VCHAR(40)
2. Name: Type of Career Service
3. Value: This will be the Customer Activity of the Person during the report period. The value will be the decoded Customer Activity description. If there are more than one Customer Activity during the report period, each Customer Activity will be listed as a separate record in the Service file. (Refer to Appendix 10 for a technical POC)

- iv. **Fieldname: TTS** (DED #036 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 4). Per meeting with CWDB on 3/22/2018, CWDB did not want CalSAWS to map the Participant Customer Activities into one of their Training Services values. Instead, the Customer Activity data will be provided as “actual values” to CAAL-Skills under DED #036 and CWDB will work with CDSS to map the Customer Activities. In light of these changes, the Data Type has changed from an INT to a VCHAR(40).**)

1. Data Type: VCHAR(40)
2. Name: Type of Training Service
3. Value: This will be the Customer Activity of the Person during the report period. The value will be the decoded Customer Activity description. If there are more than one Customer Activity during the report period, each Customer Activity will be listed as a separate record in the Service file. (Refer to Appendix 10 for a technical POC)

- v. **Fieldname: ETSR** (DED #038 of CAAL-Skills Data Exchange Requirements. **This is a deviation from the CAAL-Skills**

Data Exchange Requirement (see Supporting Documents 3). Per meeting with CWDB on 2/19/2019, CWDB agreed that the Data Type of DED #038 should be a VCHAR(5) since dynamic data types may be problematic.)

1. Data Type: VCHAR(5)
2. Name: Employment and Training Service Related to Supplemental Nutritional Assistance Program (SNAP E&T) or Welfare to Work
3. Value: This will be the CFET or WT indicator. If the Customer Activity is linked to a CW or WT program, then the value will be set to '2' – Welfare to Work. If the Customer Activity is linked to a CF or CFET program, then the value will be set to '1' – SNAP E&T. (Refer to Appendix 10 for a technical POC

vi. **Fieldname: FDA (This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 4). Per meeting with CWDB on 3/22/2018, CWDB wanted to list out the actual decoded values of all Customer Activities along with the effective dating of the activities. This data element is new and is not in the CAAL-Skills Data Exchange Requirement.)**

1. Data Type: DATE
2. Name: First Day of Activity
3. Value: This will be the begin date of the Customer Activity. (Refer to Appendix 10 for a technical POC)

vii. **Fieldname: LDA (This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 4). Per meeting with CWDB on 3/22/2018, CWDB wanted to list out the actual decoded values of all Customer Activities along with the effective dating of the activities. This data element is new and is not in the CAAL-Skills Data Exchange Requirement.)**

1. Data Type: DATE
2. Name: Last Day of Activity
3. Value: This will be the end date of the Customer Activity. (Refer to Appendix 10 for a technical POC)

5. Create a CSV file for the Supportive Services file. The population for this file will be all the Supportive Services that were provided to the Persons from the Participant file during the report period.

a. Filename: CS1_06302017_SUPP_001_R01.CSV

i. **Fieldname: PUID (DED #015 of CAAL-Skills Data Exchange Requirements. This is a deviation from the CAAL-Skills Data Exchange Requirement (see Supporting Documents 3). This element was added in as a foreign key after talking with CWDB to link records from the SUPP file to the PART file.)**












1. Data Type: VCHAR(20)
 2. Name: Participant Unique ID
 3. Value: This will be the Person's CIN. If the Person does not have a CIN, a '99' will be populated instead. (Refer to Appendix 9 for a technical POC)
- ii. **Fieldname: TOSS** (DED #064 of CAAL-Skills Data Exchange Requirements.)
1. Data Type: VCHAR(80)
 2. Name: Type of Supportive Service or Activity
 3. Value: This will be the Supportive Service that the Person received. The value will be the decoded Supportive Service the Person received during the report period with a "CALWORKS" prefix if the Supportive Service was for a CW or WT program or a "CALFRESH" prefix if the Supportive Service was for a CF or FT program. For all other programs, the Supportive Service will be prefix with "OTHER" (Refer to Appendix 11 for a technical POC). Note: The following Supportive Services will be excluded:
 - LIHEAP
 - SUAS
 - Infant Supplemental Benefit
 - State Supplemental Clothing Allowance
 - Emergency Clothing Allowance
 - Clothing Allowance
 - Education Travel Reimbursement
 - Rate Patch
- iii. **Fieldname: FDSS** (DED #065 of CAAL-Skills Data Exchange Requirements.)
1. Data Type: DATE
 2. Name: First Day of Supportive Service
 3. Value: This will be the begin date of the Service Arrangement of the Supportive Service. (Refer to Appendix 11 for a technical POC)
- iv. **Fieldname: LDSS** (DED #066 of CAAL-Skills Data Exchange Requirements.)
1. Data Type: DATE
 2. Name: Last Day of Supportive Service
 3. Value: This will be the end date of the Service Arrangement of the Supportive Service. (Refer to Appendix 11 for a technical POC)
6. Coordinate with CDSS to transfer all data files via Secure File Transfer Protocol (SFTP) per the requirements specified in the CAAL-Skills Data Exchange Requirements and MOU (see Supporting Document 1 and 2).

2.1.3 Counties Impacted

Napa County, Monterey County, and Stanislaus County

DRAFT

3 SUPPORTING DOCUMENTS

Number	Functional Area	Description	Attachment
1	N/A	CalSAWS and CDSS MOU Agreement	 CalACES_CAALSkills_MOU.DOCX
2	N/A	CAAL-Skills Data Exchange Requirements	 Attachment B - CAAL-Skills Data Exc
3	N/A	CWDB Meeting #1 Meeting Minutes	 CAAL-Skills Minutes 01.msg
4	N/A	CWDB Meeting #2 Meeting Minutes	 CAAL-Skills Minutes 02.msg
5	N/A	Email from CWDB confirming deviations from Data Exchange Requirements	 CAAL-Skills Format Change Confirmatio
6	N/A	Email from CWDB confirming Aid Code to Single Parent mappings	 RE CAAL-Skills DED #029 - Participant Si
7	N/A	Mock Files Sent to CWDB	  CS1_06302017_AAP_A_001_R01.csv CS1_06302017_ENR_O_001_R01.csv   CS1_06302017_PART_001_R01.csv CS1_06302017_SER_V_001_R01.csv  CS1_06302017_SUP_P_001_R01.csv

4 APPENDIX

1. POC: Program Person Population

```
--all CW and CF persons effective for the date range and applicable counties
create table scr_102444_pp as
(select /*+ full(ppd) */
  distinct
  pp.pers_id
  ,first_value(case.county_code) over (partition by pp.pers_id order by
ppd.end_date desc) county_code --get the last county the person was AC
  from pgm_pers pp
  ,pgm_pers_detl ppd
  ,pgm
  ,case
  where pp.id = ppd.pgm_pers_id
  and pp.pgm_id = pgm.id
  and pgm.case_id = case.id
  and case.county_code in ('27','28','50')
  and pgm.pgm_code in ('CW','FS')
  and ppd.stat_code = 'AC'
  and ppd.role_code = 'ME'
  and ppd.beg_date <= to_date('06302017','mmddyyyy')
  and ppd.end_date >= to_date('07012013','mmddyyyy')
)
/
```

2. POC: Person Address

```
--person address PH -> ML hierarchy
create table scr_102444_pa as
(select distinct
  pp.pers_id
  ,first_value(pa.pst) over (partition by pa.pers_id order by pa.type_code,
pa.end_date desc) pst
  ,first_value(pa.pcity) over (partition by pa.pers_id order by pa.type_code,
pa.end_date desc) pcity
  ,first_value(pa.ppost) over (partition by pa.pers_id order by pa.type_code,
pa.end_date desc) ppost
  from (select distinct
    pa.pers_id
    ,pa.type_code
    ,addr.city_name pcity
    ,addr.state_code pst
    ,addr.zip_code_num || addr.zip_code_suffix ppost
    ,pa.beg_date
    ,pa.end_date
  from pers_addr pa
  ,addr
  where pa.addr_id = addr.id
  and pa.beg_date <= to_date('06302017','mmddyyyy')
```

```

        and pa.end_date >= to_date('07012013','mmdyyy')
    ) pa
    ,scr_102444_pp pp
  where pp.pers_id = pa.pers_id (+)
)
/

```

3. POC: Person Race and Ethnicity

```

--person race and ethnicity
create table scr_102444_pr as
(select pr.pers_id
    ,substr(pr.ethn,1,length(pr.ethn)-1) ethn
  from (select pr.pers_id
        ,pr."01_type" || pr."02_type" || pr."04_type" || pr."05_type" ||
pr."06_type" || pr."07_type" || pr."09_type" ||
        pr."10_type" || pr."11_type" || pr."12_type" || pr."13_type" ||
pr."14_type" || pr."15_type" || pr."16_type" ||
        pr."17_type" || pr."18_type" || pr."19_type" || pr."20_type" ||
pr."cb_type" || pr."gu_type" || pr."hi_type" ||
        pr."hm_type" || pr."mx_type" || pr."oh_type" || pr."pr_type" ||
pr."sv_type" ethn
    from (select distinct
        pr.pers_id
        ,first_value(pr."01_type") over (partition by pr.pers_id order by
pr."01_type" asc) "01_type"
        ,first_value(pr."02_type") over (partition by pr.pers_id order by
pr."02_type" asc) "02_type"
        ,first_value(pr."04_type") over (partition by pr.pers_id order by
pr."04_type" asc) "04_type"
        ,first_value(pr."05_type") over (partition by pr.pers_id order by
pr."05_type" asc) "05_type"
        ,first_value(pr."06_type") over (partition by pr.pers_id order by
pr."06_type" asc) "06_type"
        ,first_value(pr."07_type") over (partition by pr.pers_id order by
pr."07_type" asc) "07_type"
        ,first_value(pr."09_type") over (partition by pr.pers_id order by
pr."09_type" asc) "09_type"
        ,first_value(pr."10_type") over (partition by pr.pers_id order by
pr."10_type" asc) "10_type"
        ,first_value(pr."11_type") over (partition by pr.pers_id order by
pr."11_type" asc) "11_type"
        ,first_value(pr."12_type") over (partition by pr.pers_id order by
pr."12_type" asc) "12_type"
        ,first_value(pr."13_type") over (partition by pr.pers_id order by
pr."13_type" asc) "13_type"
        ,first_value(pr."14_type") over (partition by pr.pers_id order by
pr."14_type" asc) "14_type"
        ,first_value(pr."15_type") over (partition by pr.pers_id order by
pr."15_type" asc) "15_type"

```

```

        ,first_value(pr."16_type") over (partition by pr.pers_id order by
pr."16_type" asc) "16_type"
        ,first_value(pr."17_type") over (partition by pr.pers_id order by
pr."17_type" asc) "17_type"
        ,first_value(pr."18_type") over (partition by pr.pers_id order by
pr."18_type" asc) "18_type"
        ,first_value(pr."19_type") over (partition by pr.pers_id order by
pr."19_type" asc) "19_type"
        ,first_value(pr."20_type") over (partition by pr.pers_id order by
pr."20_type" asc) "20_type"
        ,first_value(pr."cb_type") over (partition by pr.pers_id order by
pr."cb_type" asc) "cb_type"
        ,first_value(pr."gu_type") over (partition by pr.pers_id order by
pr."gu_type" asc) "gu_type"
        ,first_value(pr."hi_type") over (partition by pr.pers_id order by
pr."hi_type" asc) "hi_type"
        ,first_value(pr."hm_type") over (partition by pr.pers_id order by
pr."hm_type" asc) "hm_type"
        ,first_value(pr."mx_type") over (partition by pr.pers_id order by
pr."mx_type" asc) "mx_type"
        ,first_value(pr."oh_type") over (partition by pr.pers_id order by
pr."oh_type" asc) "oh_type"
        ,first_value(pr."pr_type") over (partition by pr.pers_id order by
pr."pr_type" asc) "pr_type"
        ,first_value(pr."sv_type") over (partition by pr.pers_id order by
pr."sv_type" asc) "sv_type"

```

```

from (select distinct
pr.pers_id
,(case when pr.race_type_code = '01'
then pr.short_decode_name || ','
else null
end
) "01_type"
,(case when pr.race_type_code = '02'
then pr.short_decode_name || ','
else null
end
) "02_type"
,(case when pr.race_type_code = '04'
then pr.short_decode_name || ','
else null
end
) "04_type"
,(case when pr.race_type_code = '05'
then pr.short_decode_name || ','
else null
end
) "05_type"

```

```

,(case when pr.race_type_code = '06'
      then pr.short_decode_name || ' '
      else null
     end
) "06_type"
,(case when pr.race_type_code = '07'
      then pr.short_decode_name || ' '
      else null
     end
) "07_type"
,(case when pr.race_type_code = '09'
      then pr.short_decode_name || ' '
      else null
     end
) "09_type"
,(case when pr.race_type_code = '10'
      then pr.short_decode_name || ' '
      else null
     end
) "10_type"
,(case when pr.race_type_code = '11'
      then pr.short_decode_name || ' '
      else null
     end
) "11_type"
,(case when pr.race_type_code = '12'
      then pr.short_decode_name || ' '
      else null
     end
) "12_type"
,(case when pr.race_type_code = '13'
      then pr.short_decode_name || ' '
      else null
     end
) "13_type"
,(case when pr.race_type_code = '14'
      then pr.short_decode_name || ' '
      else null
     end
) "14_type"
,(case when pr.race_type_code = '15'
      then pr.short_decode_name || ' '
      else null
     end
) "15_type"
,(case when pr.race_type_code = '16'
      then pr.short_decode_name || ' '
      else null
     end
) "16_type"

```

```

end
) "16_type"
,(case when pr.race_type_code = '17'
      then pr.short_decode_name || ' '
      else null
end
) "17_type"
,(case when pr.race_type_code = '18'
      then pr.short_decode_name || ' '
      else null
end
) "18_type"
,(case when pr.race_type_code = '19'
      then pr.short_decode_name || ' '
      else null
end
) "19_type"
,(case when pr.race_type_code = '20'
      then pr.short_decode_name || ' '
      else null
end
) "20_type"
,(case when pr.race_type_code = 'CB'
      then pr.short_decode_name || ' '
      else null
end
) "cb_type"
,(case when pr.race_type_code = 'GU'
      then pr.short_decode_name || ' '
      else null
end
) "gu_type"
,(case when pr.race_type_code = 'HI'
      then pr.short_decode_name || ' '
      else null
end
) "hi_type"
,(case when pr.race_type_code = 'HM'
      then pr.short_decode_name || ' '
      else null
end
) "hm_type"
,(case when pr.race_type_code = 'MX'
      then pr.short_decode_name || ' '
      else null
end
) "mx_type"
,(case when pr.race_type_code = 'OH'

```

```

        then pr.short_decode_name || ' ';
        else null
        end
    ) "oh_type"
    ,(case when pr.race_type_code = 'PR'
        then pr.short_decode_name || ' ';
        else null
        end
    ) "pr_type"
    ,(case when pr.race_type_code = 'SV'
        then pr.short_decode_name || ' ';
        else null
        end
    ) "sv_type"
from (select pp.pers_id
      ,pr.race_type_code
      ,cd.short_decode_name
      from pers_race pr
      ,scr_102444_pp pp
      ,(select cd.code_num_identif
        ,cd.short_decode_name
        from code_detl cd
        where cd.catgry_id = 2059
        and sysdate between cd.beg_date and cd.end_date
        )cd
      where pr.pers_id (+) = pp.pers_id
      and pr.race_type_code = cd.code_num_identif (+)
    ) pr
) pr
)
/

```

4. POC: Person Homeless

```

--person homeless
create table scr_102444_la as
(select distinct
  pp.pers_id
  ,first_value(la.type_code) over (partition by pp.pers_id order by
  la.end_date) type_code
from (select distinct
  la.pers_id
  ,la.type_code
  ,la.end_date
from living_argmt la
where la.beg_date <= to_date('06302017','mmdyyyy')
and la.end_date >= to_date('07012013','mmdyyyy')
) la
) la

```



```

        ,scr_102444_pp pp
    where pp.pers_id = la.pers_id (+)
    )
    /

```

5. POC: Person Medical Condition

```

--person disability
create table scr_102444_mc as
(select distinct
    mc.pers_id
    ,first_value(mc.disa) over (partition by mc.pers_id order by mc.disa asc) disa
from (select distinct
    pp.pers_id
    --Note: The medical condition mapping to Perm/Temp/NA/Unknown
    Disability was per the C-IV MC BA's assistance
    ,(case when mc.type_code in ('01','02','03','04','29','30','31')
        then 1 --perm disable
        when mc.type_code in
('05','06','07','08','09','10','11','12','13','14','15','16','17','18','19','20','21','22','23','24','25','26'
,'27','33','34')
        then 2 --temp disable
        when mc.type_code in ('28')
        then 98 --n/a
        when mc.type_code in ('32')
        then 90 --unknown
        when mc.type_code is null
        then 0 --not disable
    end
    )
    disa
from (select distinct
    mc.pers_id
    ,mc.type_code
    ,cd.short_decode_name disability
    ,mc.beg_date
    ,mc.end_date
    from med_cond mc
    ,code_detl cd
    where mc.type_code = cd.code_num_identif
    and cd.catgry_id = 160
    and sysdate between cd.beg_date and cd.end_date
    and mc.beg_date <= to_date('06302017','mmddyyyy')
    and mc.end_date >= to_date('07012013','mmddyyyy')
    )mc
    ,scr_102444_pp pp
    where pp.pers_id = mc.pers_id (+)
    )mc
    )
    /

```

6. POC: Person Former Foster Care Youth

```
--persons former or current FC
create table scr_102444_fc_pp as
(select distinct
  pp.pers_id
  ,(case when fc_pp.pers_id is null
    then 0
    when fc_pp.pers_id is not null
    then 1
    else 99
  end
)
  fctrc
from (select distinct
  pp.pers_id
  from pgm_pers pp
  ,pgm
  where pgm.id = pp.pgm_id
  and pgm.pgm_code = 'FC'
)fc_pp
,scr_102444_pp pp
where fc_pp.pers_id (+) = pp.pers_id
)
/
```

7. POC: Person Migrant Farm Worker

```
--person migrant worker
create table scr_102444_mg_pp as
(select distinct
  pp.pers_id
  ,(case when mg_pp.pers_id is null
    then 0
    when mg_pp.pers_id is not null
    then 1
    else 99
  end
)
  farm
from (select /*+ full(res) */
  res.pers_id
  ,res.migr_seasnl_farm_wrkr_ind farm
  from res
  where res.migr_seasnl_farm_wrkr_ind = 'Y'
  and res.beg_date <= to_date('06302017','mmddyyyy')
  and res.end_date >= to_date('07012013','mmddyyyy')
) mg_pp
,scr_102444_pp pp
where pp.pers_id = mg_pp.pers_id (+)
)
/
```

```

/
8. POC: Person Single Parent
create table scr_102444_sp as
(select distinct
  pp.pers_id
  ,ac.aid_code
  ,(case when ac.aid_code in
('33','35','3C','3F','3G','3H','3M','3U','4N','4R','R5','R8')
  then 0
  when ac.aid_code in
('09','0F','30','32','3A','3E','3L','3W','4P','K1','R4','R6','R7','R9')
  then 99
  else 98
  end
  )
  )
spar
from (select /*+ full(pgm_pers) full(ac) */
  distinct
  pgm_pers.pers_id
  ,first_value(ac.aid_code) over (partition by pgm_pers.pers_id order by
ac.end_date desc, pgm.pgm_code asc) aid_code --get latest assigned CF or
CW Aid Code. If tie, get CW Aid Code
  from pgm_pers
  ,pgm
  ,case
  ,aid_code ac
  where pgm_pers.pgm_id = pgm.id
  and pgm.case_id = case.id
  and pgm_pers.pgm_id = ac.pgm_id
  and case.county_code in ('27','28','50')
  and pgm.pgm_code in ('CW','FS')
  and ac.beg_date <= to_date('06302017','mmdyyyy')
  and ac.end_date >= to_date('07012013','mmdyyyy')
  ) ac
  ,scr_102444_pp pp
  where pp.pers_id = ac.pers_id (+)
  )
/

```

```

9. POC: Person Participant (ALL)
--PART file
create table scr_102444_part as
(select distinct
  pp.county_code eoid
  ,pers.cin_num_identif puid
  ,(case when pers.ssn is not null
  then pers.ssn
  else '99'
  end
  )
  )
end

```

```

) ssn
,trim(upper(pers.first_name)) pfnm
,trim(upper(pers.mid_name)) pmnm
,trim(upper(pers.last_name)) plnm
,to_char(pers.dob,'mmdyyy') dob
,(case when pers.gender_code = 'MA'
      then 'M'
      when pers.gender_code = 'FE'
      then 'F'
      else null
      end
) sex
,trim(upper(pa.pcity)) pcity
,trim(upper(pa.pst)) pst
,pa.ppost
,(case when ms.pers_id is not null
      then 1
      else 0
      end
) vet
,trim(upper(pr.ethn)) ethn
,' ' race --IMPORTANT PLEASE READ: This value will always be null, however
when creating a new table there cannot be a null or blank value.
--The ' | ' needs to be converted to a ' || ' (null) once the data has
been extracted to a flat file
,(case when la.type_code is null
      then '0'
      when la.type_code is not null
      then '1'
      else '99' --unknown
      end
) hmls
,fc_pp.fctr
,mc.disa
,mg_pp.farm
,sp.spar
from scr_102444_pp pp
,pers
,scr_102444_pa pa
,mil_srv ms
,scr_102444_pr pr
,scr_102444_la la
,scr_102444_mc mc
,scr_102444_fc_pp fc_pp
,scr_102444_mg_pp mg_pp
,scr_102444_sp sp
where pp.pers_id = pers.id
and pp.pers_id = pa.pers_id (+)

```

```

and pp.pers_id = ms.pers_id (+)
and pp.pers_id = la.pers_id (+)
and pp.pers_id = mc.pers_id (+)
and pp.pers_id = pr.pers_id (+)
and pp.pers_id = fc_pp.pers_id (+)
and pp.pers_id = mg_pp.pers_id (+)
and pp.pers_id= sp.pers_id (+)
)
/

```

10. POC: Person Services (ALL)

```

--SERV file
create table scr_102444_serv as
(select pers.cin_num_identif puid
,(case when ca.pgm_code in ('CW','WT')
then 7
when ca.pgm_code in ('FS','FT')
then 8
else 99
end
) fsrc
,trim(upper(ca.type_code)) tsc
,trim(upper(ca.type_code)) tts
,(case when ca.pgm_code in ('CW','WT')
then 2
when ca.pgm_code in ('FS','FT')
then 1
when ca.pgm_code is null
then null
else 0
end
) etsr
,to_char(ca.beg_date,'mmdyyy') fda
,to_char(ca.end_date,'mmdyyy') lda
from (select /*+ full(ca) */
ca.pers_id
,pgm.pgm_code
,cd.short_decode_name type_code
,ca.beg_date
,ca.end_date
from cust_activ ca
,pgm_cust_activ pca
,pgm_pers pp
,pgm
,case
,(select cd.code_num_identif
,cd.short_decode_name
from code_detl cd
where cd.catgry_id = 56

```

```

        and sysdate between cd.beg_date and cd.end_date
    ) cd
where ca.id = pca.cust_activ_id
and pca.pgm_pers_id = pp.id
and pp.pgm_id = pgm.id
and pgm.case_id = case.id
and ca.type_code = cd.code_num_identif
and case.county_code in ('27','28','50')
and pgm.pgm_code in ('CW','FS','WT','FT')
and ca.beg_date <= to_date('06302017','mmddyyyy')
and ca.end_date >= to_date('07012013','mmddyyyy')
) ca
,scr_102444_pp pp
,pers
where pp.pers_id = pers.id
and pp.pers_id = ca.pers_id (+)
)
/

```

11. POC: Person Supportive Services (ALL)

```

create table scr_102444_supp as
(select pers.cin_num_identif puid
,trim(upper(supp.toss)) toss
,to_char(supp.fdss,'mmddyyyy') fdss
,to_char(supp.ldss,'mmddyyyy') ldss
from (select /*+ full(sa) */
distinct
cn.pers_id
,cn.type_code
,(case when pgm.pgm_code in ('CW','WT')
then 'CALWORKS ' || cd.short_decode_name
when pgm.pgm_code in ('FS','FT')
then 'CALFRESH ' || cd.short_decode_name
else 'OTHER ' || cd.short_decode_name
end
) toss
,sa.period_beg_date fdss
,sa.period_end_date ldss
from cust_need cn
,serv_arrgmt sa
,serv_arrgmt_detl sad
,pgm
,(select cd.code_num_identif
,cd.short_decode_name
from code_detl cd
where cd.catgry_id = 164
and sysdate between cd.beg_date and cd.end_date
) cd
where cn.id = sa.cust_need_id

```

```
and sa.id = sad.serv_arrgmt_id
and pgm.id = sa.pgm_id
and cn.type_code = cd.code_num_identif
and sa.period_beg_date <= to_date('06302017','mmddyyyy')
and sa.period_end_date >= to_date('07012013','mmddyyyy')
and sad.stat_code = 'AP'
and cn.type_code not in ('LH','SU','IS','SS','ER','RP','EC','CA') --exclude these
since they are not supportive services for CW, CF, WT, and FT
) supp
,scr_102444_pp pp
,pers
where pp.pers_id = pers.id
and pp.pers_id = supp.pers_id (+)
)
/
```

DRAFT