Document Attributes

| Attribute | Value |
| --- | --- |
| Application ID /  Application Name | OPUS/OM |
| Owner | Arun Kumar Madas |
| Owner Contact Information | Am6489 |
| Other Attribute |  |

## 

Revision History

The following table lists the revision history of this document:

| Author | Date | Version # | Revision Description |
| --- | --- | --- | --- |
| Arun Kumar Madas | 1/5/2016 | 0.1 | Initial Version.  Added initial design elements. |
| Arun Kumar Madas | 1/25/2016 | 0.2 | Updates after HLD review call |

Table of Contents

[Overview 3](#_Toc374017362)

[Problem Statement 3](#_Toc374017363)

[Design Decisions 3](#_Toc374017364)

[Alternative Designs 33](#_Toc374017365)

[Assumptions/Risks 33](#_Toc374017366)

[Pre-Production Disaster Recovery Planning 33](#_Toc374017367)

[Other Plans and References 33](#_Toc374017368)

[Acceptance & Approvals 33](#_Toc374017369)

## Overview

The High Level Design (HLD) describes how an application will implement the architectural concept and meet the requirements allocated to that application. The High Level Design describes the functionality the application will provide, the overall design for providing that functionality and meeting the nonfunctional requirements, and the rationale for choosing that design. The High Level Design also describes how the application will cooperate and interface with other applications to provide an integrated solution that achieves the architectural concept.

The High Level Design covers these topics:

* Problem Statement
* Design Decisions
* Alternative Designs
* Assumptions/Risks
* Other Plans/References

**In-Scope**

* OPUS, OPUS Mobile (OM)
* Channels and customer type:
  + All flows, channels and customer types that impacted NEXT implementation.
* NEXT offer type(/financing plan type): NE
* Functionalities:
  + **Next Upgrade (trade-in) eligibility based on percentage payment completed**- As of today, in production the upgrade eligibility for a given Next plan is calculated based on the number of installment payment completed. Going forward, with this project implementation the Next upgrade (trade-in) eligibility is expected to be calculated based on the percentage payment completed of the NEXT device price.

For example:

As of today in production- for a NEXT 18/24 plan, customer will be eligible to trade-in after making 18 installment payments.

Going forward, with the new % based NEXT plans that will be introduced with this project, the customer will be eligible to trade in on completion of payment of X% (configurable by business in TLG/MRE) of the NEXT retail price of the device.

* + - * OPUS shall be enhanced to:
  + Support the display and selection of the new % based NEXT plans as sent by TLG/MRE.
  + Ensure that the new % based NEXT plans and the legacy plans are mutually exclusive in OPUS for display and selection, i.e., if OPUS displays the new % based NEXT plans, the legacy plans shall not be displayed and vice-versa(except for BRE flows).
  + Implement a control mechanism to filter the display the new % NEXT plans and legacy plans to support pilot launch.
* **Support flexible down payment-** As of today, in production, TLG/MRE returns some upgrade offers which require a down payment. But, the down payment value is fixed. Going forward, the customer can make down payments for any offer, thus decreasing the monthly EMI.
  + OPUS shall be enhanced to:
    - Be able to accept a down payment as needed by the customer for the new % model plans
    - Implement a logic to limit the customer from making a down payment more than the allowable maximum down payment value and less than the minimum allowed.

**Out of Scope**

* The following customer groups will not be supported by this project:
  + Apple does not use OPUS for NEXT
* Digital Life Customer
* PrePaid
* Resellers such as GoPhones/Track phones
* % based trade-in model will not be supported for EIP plans (i.e plans with offer type= EIP)

## Risk

### R-1 TLG/MRE uses Market level flag to filter the new % model NEXT offers, while OPUS uses the store level flags to filter the new % model NEXT offers. If there is a mismatch between the market level flag and the store level flag in that market, customer would see different eligibility in different stores during Pilot launch.

~~For example if Dallas market level flag (used by TLG/MRE) is turned ON for Pilot launch, and one of the stores in Dallas market has the store level flags set to OFF. In this case, MRE would return New NEXT plan to OPUS, and OPUS will not display this offer to the customer since the Pilot flag is OFF in the store. However, if the same customer goes to a different store in the same market, but with the store flag ON, OPUS will display the offer to the customer.~~

### ~~Mitigation: Business shall make sure that the store flags shall match with the market level flags i.e., if Dallas market level flag (used by TLG/MRE) is turned ON for Pilot launch, all the store level flags (used by OPUS) in the Dallas market shall be ON.~~

## 

## Problem Statement

Primary objective is to simplify the offer mix for customers on AT&T Next while expanding flexibility at the same time.

* We accomplish this by switching to a “percentage model,” whereby the trade-in amount is taken as a percentage of the Next sales price, which as a point of reference will be the “No Commitment Price.”
* A new Next with minimum down payment offer will be developed as part of this project. This will also be based off percentage amounts. This offer is necessary to enable sales extending Next to credit challenged customers.

## Design Decisions

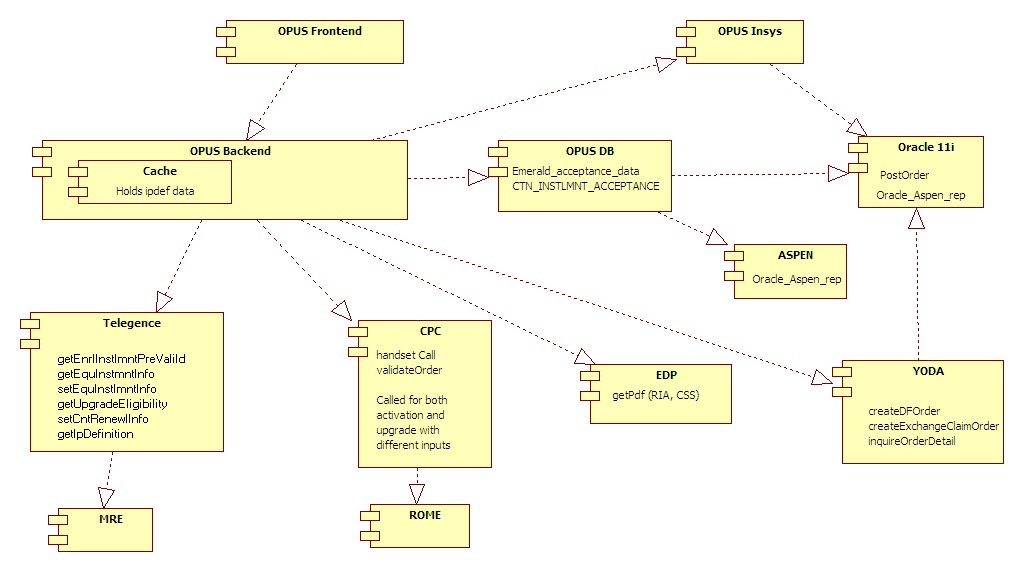
|  |  |  |
| --- | --- | --- |
| **Req. ID** | **Design Element** | **Trace-To** |
| 288312\_OPUS\_HLD\_010 | OPUS shall be enhanced to be able to display the new percentage model NEXT offers as returned by TLG/MRE.    Refer wireframes for presentation details.  <http://clddev0srv04830:8600/OPUS/OPUS_Launcher_DEV/pdfs/OPUS/OPUSMobile_288312_Next2FlexibleDownPayments.pdf>  OPUS will either display legacy next plans or new Next plans depending on the project flag but not both the plans at the same time except in BRE flow.  Files to Modify:   1. FinancePlansPartial.html 2. NewUpgradeEligibilityResults.jsp 3. OMNewUpgradeEligibilityResults.jsp 4. OMNewTitanUpgradeEligibilityResults.jsp 5. ChangeContract.jsp 6. OMChangeContract.jsp 7. OMTitanChangeContract.jsp | 288312\_OPUS\_SR\_010 |
| 288312\_OPUS\_HLD\_020 | OPUS shall pass the USC code in the ‘Channel’ field while calling TLG API to validate installment eligibility.  Currently OPUS is not passing the USC code in the channel field while calling TLG API to retrieve the Next plans. OPUS will use the same USC code retrieve method like upgrades to get the USC code for each chhanel and will pass it to TLG. OPUS will be getting this value from the store flag MRE\_UPGRADE\_SECURITY\_CODE. Following is the current USC code setup for different channels.  TLG API: Appquey. getEnrlInstlmntPreValid  API input field: instlmntPreValidAddInfo.channel  [Sample values from QC DB are shown below]    Files to Modify:   1. TLGGenericQryAndUpdCmd.java   Modify the existing method checkEmeraldEligibility to get USC code from storeprofile and pass to TLG in prevalidInfo.channel field. | 288312\_OPUS\_SR\_015 |
| 288312\_OPUS\_HLD\_030 | OPUS shall have the ability to turn ON/OFF the functionality of this project at the store level.   * If the flag is set to **ON**, OPUS shall offer all the functionalities as stated in this document. * If the flag is set to **OFF**, OPUS shall not offer any functionality as stated in this document.   OPUS shall have the ability to control the display of the new % model NEXT offers returned by TLG/MRE at a store level to support pilot launch.  If the OPUS flag is ON, then OPUS shall display only the new % model NEXT plans.  Note:   * The current permissions that allow NEXT ordering shall be re-used/ continued to be used.   Assumption:   * OPUS cannot filter/control display of each offer within the same model. OPUS will only have the control to display of the offers of new model(%), i.e., if TLG/MRE sends 3 ‘percentage’ offers and 2 legacy offers, OPUS cannot hide 1 ‘percentage’ offer and display the other 2 percentage offers.   TLG/MRE is expected to filter out any % offers which must not be presented to the customer.  This requirement does not apply to BRE flows. In a BRE flow, OPUS can display both legacy and new NEXT offers to the customer as returned by MRE.  OPUS will create a new store level flag for this project to support the new next plans and flexible downpayments.  Flag Name: SUPPORT\_NEWNEXT\_FLEXIBLEDP\_PLANS  For example,  If TLG/MRE returns 3 percentage model offers and 2 legacy NEXT plans, if the OPUS flag for percentage model offers is OFF, then OPUS shall display only the legacy NEXT plans for selection as returned by TLG/MRE and when the OPUS flag for percentage model offers is ON, OPUS will display the new NEXT plans for selection as returned by TLG/MRE.  OPUS will consider the plan as legacy NEXT if the upgradePerc field is not greater then zero and this applicable only to offer/plan type “NE” and will not filter for any other offer/plan type. BASICQIP20  will not be filtered and OPUS will depend on current enterprise config value for this**.**  OPUS will add an additional filtering logic to display only non DP plans when TLG/MRE returns both DP and non DP and is applicable for flexible downpayment allowed plans and flexible downpayment flag is ON.  OPUS will create a enterprise config to store the flexible downpayment ineligible plans types.   |  |  |  |  | | --- | --- | --- | --- | | **Plan type eligible for Flexible DP** | **Both DP and non DP returned** | **Only DP plans returned** | **Only non DP plans retuned** | | Yes | Display Only non DP | Display DP plans | Display non DP | | NO | Display both DP and non DP | Display DP plans | Display non DP |   Activations:  TLG API: AppQuery. getEnrlInstlmntPreValid  API output field: upgradePerc field in IpDefinitionOutInfo object  Upgrades:  TLG API: AppQuery. getUpgradeEligibility  API output field: upgradePerc field in UpgEligResInfo object  The following table shows the display of new/legacy plans in each flow with the OPUS store level flag ON and OFF.    Files to Modify:   1. WirelessAccountManager.java   Create a new Method isPercNextandFlexibleDPFlagOn to check the flag value and return a Boolean. Follow isShowUpdatedNextEliMsg for reference.  Modify the existing method getUpgradeEligibilityResults to check for the isPercNextandFlexibleDPFlagOn and set the flexibleDownpaymentEnabled in UpgradeEligibilityRequest object.   1. TLGGenericQryRequestInfo.java   Create a new boolean percNextandFlexibleDPEnabled and setter and getter for it.   1. CustomerAccountManager.java   Modify the existing method checkemeraldeligibilty(HttpServletRequest request, String ban, String market, int flowType, boolean toBRMode, String planType, char mode, boolean nextFlexibleLineLimitFlow, boolean checkMaxAllowed,String imeiType) to check the isPercNextandFlexibleDPFlagOn method and set the percNextandFlexibleDPEnabled variable to filter the plans in the backend.   1. NextEligibilityPlanInfo.java   Create new Money object minFinanceAmount, Boolean allowflexibleDowpayment and setter and getter methods.   1. TLGGenericQryAndUpdCmd.java   Modify the existing checkEmeraldEligibility method to filter the finance plans returned by TLG. If the percNextandFlexibleDPEnabled then add the plans with plan type ‘NE’ only if upgradePerc >0 and not add the legacy plans to financePlans object that being created since both NEW next and legacy can not co-exist. The logic for all other plantypes is BAU. In NextEligibilityPlanInfo object set the minFinanceAmount returned by TLG and allowflexibleDowpayment from percNextandFlexibleDPEnabled TLGGenericQryRequestInfo.  OPUS will add an additional logic to check if both DP and non DP plans were returned for each plan type when the percNextandFlexibleDPEnabled is enabled and the plantype is allowed for flexible downpayment and will return only non DP plans when both DP and non DP plans were returned by TLG/MRE.   1. UpgradeQualificationLevel.java   Create new Money object minFinanceAmount, String tradeInUpgradePerc, Boolean allowflexibleDowpayment and setter and getter methods.   1. TLGEquipmentUpgradeQryCmd.java   Modify existing populateQualificationLevelDetails method to set the new minFinanceAmount, tradeInUpgradePerc values returned by TLG.   1. POSEquipmentUpgrade.java   Create new Boolean legacyContract and setter and getter methods.   1. WirelessAccountManager.java   Modify the existing populatePlanFriendlyNameNStoreflag method to filter the new NEXT and legacy Next plans based on the isPercNextandFlexibleDPFlagOn flag and flowtype[upgrade, BRE].  Mofidy the existing buildEquipmentInstallmentInfo method to call ResourceManagerImpl. islegacyContract method by passing the plandientifier of the current finance plan and set the value in POSEquipmentUpgrade. legacyContract  If(isPercNextandFlexibleDPFlagOn == false)  {  // add only legacy next[upgradePerc not > 0] and other plans[IP,SP,EIP]  }else if(isPercNextandFlexibleDPFlagOn == true )  {  If(regular upgrade)  {  // add only NEW next[upgradePerc > 0] and other plans[IP,SP,EIP, basicQIP20]  }  else if(BRE)  {  // add NEW plans and other plans[IP,SP, EIP] and legacy plans also only when the current contract is legacy.  }  }  OPUS will add an additional logic to check if both DP and non DP plans were returned for each plan type when the flexible downpayment is enabled and will return only non DP plans when both DP and non DP plans were returned by MRE.   1. ResourceManagerImpl.java   Create a new method islegacyContract(String currentPlanID) to return a Boolean after comparing the currentplanidentifier with IPdefiniton planidentifier.   1. TitanMobileServiceAction.java   Mofidy the existing filterPOSEquipmentUpgrade method to call ResourceManagerImpl. islegacyContract method by passing the plandientifier of the current finance plan and set the value in POSEquipmentUpgrade. legacyContract | 288312\_OPUS\_SR\_020  288312\_OPUS\_SR\_030  288312\_OPUS\_SR\_170 |
| 288312\_OPUS\_HLD\_040 | When a customer opts to perform a BRE transaction, OPUS shall:   1. Display only legacy offers, if: 2. Store flag is OFF 3. Display only new NEXT offers, if: 4. Store flag is ON and customer was on a New NEXT plan or subsidy while coming in for a BRE 5. Display both legacy and new offers (as returned by MRE), if:   Store flag is ON and customer was on a legacy plan while coming in for a BRE  OPUS will determine if the customer is on legacy plan using the planIdentifier field in equipment installment details returned by TLG for that subscriber and compare it with the planIdentifier field in getIpDefinition and checks for upgradePerc field value. If the upgradePerc is not greater than zero and the plan type is ‘NE’ then it will be considered as legacy plan.  TLG API’s: Appquery. getEquInstmntInfo , Refquery. getIpDefinition  **BASICQIP20 plan will not be condisered as legacy NEXT plan.** | 288312\_OPUS\_SR\_040 |
| 288312\_OPUS\_HLD\_050 | When a customer does a NEXT activation/upgrade transaction, OPUS shall be enhanced to persist and pass the following information to TLG:   * Price of device (i.e. Next Retail Price) * Down payment information (total DP amount = voluntary + involuntary) * Total amount financed   OPUS will be enhanced to update the downpayment amount to include both the voluntary + involuntary amount.  Files to Modify:   1. PlanData.java   Create a new Money object selectedDPAmt, bmgSelectedDPAmt and setter and getter method.  Modify the existing getCalculatedDownPaymentAmt, getCalculatedBMGDownPaymentAmt methods to return the selected DP amount when percent Next and flexible Downpayment flag is ON  Set the selectedDPAmt, bmgSelectedDPAmt[for bmg customers] in PlanData object from the amount user entered.  Create a new method requiredminDPAmt to calculate the required down payment amount based on the downpayment percentage returned by TLG.   1. POSEquipmentUpgrade.java   Create a new Money object selectedDPAmt, bmgSelectedDPAmt and setter and getter method.  Modify the existing getCalculatedDownPaymentAmt, getCalculatedBMGDownPaymentAmt methods to return the selected DP amount when percent Next and flexible Downpayment flag is ON  Set the selectedDPAmt, bmgSelectedDPAmt[during bmg upgrade] in POSEquipmentUpgrade object from the amount user entered.  Create a new method requiredminDPAmt to calculate the required down payment amount based on the downpayment percentage returned by TLG. | 288312\_OPUS\_SR\_120 |
| 288312\_OPUS\_HLD\_060 | OPUS shall be enhanced to be able to collect a down payment from the customer as per their choosing (voluntary or involuntary or both), when opting for an installment billing plan.  Note:   * Flexible down payment shall not be offered for Legacy NEXT offers. * Refer wireframes for presentation details.   OPUS does support the involuntary downpayment functionality now based on downpayment percentage returned by TLG and it will be enhanced to support voluntary or involuntary or both.  When the project level flag is ON, OPUS will provide the flexible downpayment option for all the NEW NEXT plans with plan type ‘NE’ and existing plans with plan type other than ‘NE’ [like SP, IP, EIP, BASICQIP20]  ~~If an plantype is ineligible for flexible downpayment based on the enterprise config flexible\_ineligible\_plantype, OPUS will not provide flexible downpayment option and follows BAU process.~~    Files to Modify:   1. PlanData.java/ UpgradeQualificationLevel.java   Create a new Boolean allowflexibleDownPayment and setter and getter methods. The values will be set in TLGGenericQryAndUpdCmd.checkEmeraldEligibility[Activation] and WirelessAccountManager.populatePlanFriendlyNameNStoreflag [upgrade].   1. FinancePlansPartial.html   If the plandata.allowflexibleDownPayment is true provide a dropdown for the user to select the voluntary downpayment. The initial value will be always the required involuntary downpayment amount and the maximum will be the maximum allowed downpayment amount. The values in the downdrop will be incremenatal of $50 from min to max downpayment amount allowed. Also provide an option for the user to enter any amount when the option selected is other.  Update the financedAmount and MonthlyPayment when user changes the dropdown value or clicks calculate button for other option.  For example:  value is the minimum down payment. Increment values in  dropdown by $50 until the maximum down payment is  reached.  Example: Minimum DP=$165, Maximum DP=$370  Values: $165, $200, $250, $300, $350, $370  Example: Minimum DP=$0, Maximum DP=$370  Values: $0, $50, $100, $ 150, $200, $250, $300, $350, $370  Max allowed Downpayment = Equipment MSRP – Minimum finance amount.  Minimum downpayment = downpaymentpercentage X Device price   1. ChangeContract.jsp/OMChangeContrcat.jsp/OMTItanChangeContract.jsp   OPUS will modify the existing contract selection screen to support voluntary downpayment selection.  If the UpgradeQualificationLevel.allowflexibleDownPayment is true provide a dropdown for the user to select the voluntary downpayment. The initial value will be always the required involuntary downpayment amount and the maximum will be the maximum allowed downpayment amount. The values in the downdrop will be incremenatal of $50 from min to max downpayment amount allowed. Also provide an option for the user to enter any amount when the oprion selected id other.  Update the financedAmount and MonthlyPayment when user changes the dropdown value or clicks calculate button for other option.  For example:  value is the minimum down payment. Increment values in  dropdown by $50 until the maximum down payment is reached.  Example: Minimum DP=$165, Maximum DP=$370  Values: $165, $200, $250, $300, $350, $370  Example: Minimum DP=$0, Maximum DP=$370  Values: $0, $50, $100, $ 150, $200, $250, $300, $350, $370     1. TLGAccountActivation.jsp/OMReviewAndActivate.jsp   UpgradeSummary.jsp/OmUpgradeSummary.jsp  OPUS will provide an option to modify the downpayment amount selection on wireless review and activate and upgrade summary screens in non unified flow. The logic should be similar to rate plan screen and update the selectedDPAmt, bmgSelectedDPAmt fields in planData objects. | 288312\_OPUS\_SR\_050 |
| 288312\_OPUS\_HLD\_070 | When a customer makes a down payment opting for an installment offer, OPUS shall calculate and display the reduced the monthly payment amount using the logic shown below:  For example,  For a device of $600 over 20 months, if the monthly payment amount is $30,  if the customer makes a down payment of $ 200, the monthly payment shall come down to $20.00($400/20)  OPUS shall be enhanced to prevent the customer from making a down payment beyond a maximum amount that is computed based on the TLG returned minimum financed amount.  Maximum allowed down payment shall be calculated as below:    *(NEXT RP-minimum finance amount)=Maximum allowed down payment*  For example,  If the NEXT retail price of a device is $600 and business configures a minimum finance amount as $240, OPUS shall not allow the customer to make a down payment of more than $360.00  Note:   * This requirement applies to Voluntary, involuntary and voluntary+ involuntary down payments * Refer wireframes for presentation details   OPUS will recalculate the finance amount, monthly payment after user enters the downpayment amount and clicks on calculate button as shown in below WF or selects a downpaymnet amount from the dropdown. The maximum allowed downpayment will be calculated using the following.  **Max allowed Downpayment = Equipment MSRP – Minimum finance amount.**  **Finance Amount = Device price – selected downpayment amount**  **Monthly Payment = finance Amount / Length of term**    OPUS will show an error message to the user if the the downpayment required does not met the minimum downpaymet required or higher than the maximum allowed downpayment. | 288312\_OPUS\_SR\_060  288312\_OPUS\_SR\_070 |
| 288312\_OPUS\_HLD\_080 | If TLG returns a minimum down payment % to be collected for the selected finance offer, OPUS shall ensure that the customer will not be able to continue without meeting the minimum down payment rule.  Note:   * The minimum down payment value shall always be zero or more.   If the customer makes a down payment greater than the minimum down payment (i.e, customer makes a voluntary down payment), OPUS is not expected to separate the voluntary and involuntary amounts before sending it to the downstream systems.  Down payment amount shall be sent as a single lump sum value.  OPUS will calculate the minimum downpayment required for a selected plan and will default it that value.  Minimum Downpayment required = %downpaymnet required X device price.  OPUS will not differentiate the voluntary and involuntary downpaymnet any where in the flow and while submitting to the downstreams.  Files to Modify:   1. PlanData.java   Create a new Money object selectedDPAmt, bmgSelectedDPAmt and setter and getter method.  Modify the existing getCalculatedDownPaymentAmt, getCalculatedBMGDownPaymentAmt methods to return the selected DP amount when flexible Downpayment flag is ON  Set the selectedDPAmt, bmgSelectedDPAmt[for bmg customers] in PlanData object from the amount user entered.  Create a new method requiredminDPAmt to calculate the required down payment amount based on the downpayment percentage returned by TLG.   1. POSEquipmentUpgrade.java   Create a new Money object selectedDPAmt, bmgSelectedDPAmt and setter and getter method.  Modify the existing getCalculatedDownPaymentAmt, getCalculatedBMGDownPaymentAmt methods to return the selected DP amount when flexible Downpayment flga is ON  Set the selectedDPAmt, bmgSelectedDPAmt[during bmg upgrade] in POSEquipmentUpgrade object from the amount user entered. | 288312\_OPUS\_SR\_080  288312\_OPUS\_SR\_090 |
| 288312\_OPUS\_HLD\_90 | If a customer opts to make a down payment, OPUS shall display the down payment amount on the printed and emailed receipt. (BAU).  This is BAU for downpayment greater than zero.  If there is a Buyback credit in the cart, OPUS shall allow the user to use the Buyback credit towards down payment.  **(BAU) TEST ONLY**  Note:   * Today, Buyback amount is added to the cart, and during ring out, the amount can be used towards the total cart due. * Only COR store are in scope for using Buyback credits. The use of Buyback gift cards for Down payment is not in scope.   OPUS shall allow customers to use any of the below tender types to make involuntary/voluntary down payments:  Support in all sales channels   * Credit Card * AT&T Gift Card * Debit Card   Support in COR, Indirect channels, National Retailers and Apple only   * Cash   Support in COR only   * Automated Clearinghouse (ACH) * Buyback Credit   Note: Acceptance of ACH tender type shall be based on BAU eligibility rules.  **This is BAU- TEST ONLY requirement** | 288312\_OPUS\_SR\_130  288312\_OPUS\_SR\_100  288312\_OPUS\_SR\_110  **TEST ONLY** |
| 288312\_OPUS\_HLD\_100 | On the NEXT offers eligibility screen, OPUS shall be enhanced to display a message notifying the user about any buyback credit that is available in the cart.  OPUS will check if any buyback transactions exists in cart and if there exists BB transactions with price greater than zero, OPUS will add the price for all the transactions and will display the credit in rate plan, activation summary[for bmg customers only] in activation flow and contract selection screen, upgrade summary[for bmg upgrade customers only] in upgrade flow.    Files to Modify:   1. InventoryManager.java   Create a new Method calculateBuyBackAmount to loop through all the transactions and check for SerializedBuyBackTransaction with amount greater than zero and return the total buyback credit.   1. ActivationForm.java/RPFDetails.java/ POSEquipmentUpgrade.java/   TitanMobileServiceForm.java/ TitanWirelessOrderForm.java  Create a new string buybackCreditMessage and setter and getter methods.   1. HandsetInfoAction.java/ TLGChangeRatePlanFeaturesAction.java/UpgradeProcessAction.java/ TitanWirelessOrderction.java/TitanMobileServiceAction.java   Call InventoryManager. calculateBuyBackAmount method and set the requied buybackCreditMessage in the required object.   1. FinancePlansPartial.html/ChangeContract.jsp/OMChangeContract.jsp   /OMTitanChangeContract.jsp  Modify the above files to display the buyback credit message when flexible Downpayment plans are available.  [OPUS will create a new enterprise\_config for this display.] | 288312\_OPUS\_SR\_105 |
| 288312\_OPUS\_HLD\_110 | OPUS shall be enhanced to pass the following to ORACLE after order completion for C&C order (in NRLDTrx and postOrder API):   * Installment IPID (planID) [planIdentifier from IP definition]   Oracle NRLDTrxSyncService and Post order APIs are being enhanced to support plan id identifier input. OPUS will modify the existing logic to pass the plan identifier for all the C&C NEXT transactions.  Files to Modify:   1. NRLDTrxServiceRequest.java   Create a new String installmentIPID and setter and getter methods.   1. SubmitActivationAction.java 2. TitanWirelessOrderAction.java 3. TitanMobileServiceAction.java 4. ChangeEquipmentAction.java 5. ReverseUpgradeProcessAction.java 6. UpgradeprocessAction.java   Modify all the above files where NRLDTrxServiceRequest is being set to pass the plan identifier also in the request.   1. FusionWebServicesManager.java   Modify processNRLDTrxService Method to set the plan identifier in the InputNRLDTrxLine request object.   1. InsysOrderLineInfo.java   Create a new String installmentIPID and setter and getter methods.   1. InsysOrderLineCreatorHandler   Modify the below methods to set installmentIPID in InsysOrderLineInfo.     1. SalesAdapter.java   Modify the method getOrderLineDetail to pass the INSTALLMENT\_IPID to oracle in postOrder API.   1. CustomerServicesDAO.java   Create a new column in NRLD\_PDC\_TMP table to store the plan identifier field. Modify the existing stored procedure P\_NRLD\_FUSION.INSERT\_NRLD\_PDC\_TMP to take one more input field and set it as plan identier and insert in NRLD\_PDC\_TMP.installment\_planidentifer.  Modify the existing insertNRLDFusionServiceStatus method to pass the plan identifier from NRLDTrxServiceRequest object.  Sales dev team will do code changes for postorder API . | 288312\_OPUS\_SR\_150 |
| 288312\_OPUS\_HLD\_120 | OPUS shall be enhanced to pass the following to YODA after order completion for a DF order (in CreateDFOrder, CreateExchangeClaimOrder):   * Installment IPID (planID)   Yoda CreateDFOrder and CreateExchangeClaimOrder APIs are being enhanced to support plan id identifier input. OPUS will modify the existing logic to pass the plan identifier for all the NEXT transactions.  Files to Modify:   1. YodaOrderLineCreatorHandler.java   Modify the existing methods getOrderLineDT, getExchangeCorReturnLineDT to set the installmentIPID in the OrderLineDT yoda object simiar to InstallmentCIPID. installmentIPID is being populated similar to InstallmentSeqID in InstallmentAgreementTender and just need to use the value.  Sample values: BASICQIP20,NXT12DP2,NXT12DP3,EIP18DP,  EIP20,EIP20DP,EIP32,NXT12A   1. RestockingFeeAction.java   Modify the existing method exchangeLocalDealer to set the planId in the returnInstTender object.   1. YodaWebServicesManager.java   Modify the existing inquireOrderDetail method to set the planId in SearchOrderResult object. | 288312\_OPUS\_SR\_160 |
| 288312\_OPUS\_HLD\_130 | OPUS shall be enhanced to pass the following to ASPEN after order completion:   * Installment IPID (planID)   Note:  The existing process will be leveraged to pass the new element to ASPEN  No Code changes. Aspen will be getting this information from the biller[TLG/Enabler] and are not excepting this from OPUS. | 288312\_OPUS\_SR\_165 |
| 288312\_OPUS\_HLD\_140 | **Modifiy BOPIS order flow**  OPUS will support new percentage and flexible downpayment Next plans when the PercNextandFlexibleDPFlagOn is enabled in the modify bopis activation and upgrade order flows.  Files to modify:   1. TLGFeaturesAction.java   Modify the existing method updatebopisorder to set the downpayment amount user selected and update the finance amount and monthly payment accordingly.   1. UpgradeProcessAction.java   Modify the existing method updatebopisorder to set the downpayment amount user selected and update the finance amount and monthly payment accordingly. | 288312\_OPUS\_SR\_070 |
| 288312\_OPUS\_HLD\_150 | **Edit DP in cart:**  OPUS shall allow the user to edit the DP amount after the installment device is added to the cart.  After the DP amount has been changed, OPUS shall ensure that the RIA is re generated and presented to the customer.  Note:   * The changed DP shall align with all the flexible down payment rules stated in this project. * This requirement applies to C&C (in COR) order only.   OPUS will modify the existing edit Next device from cart functionality to support modifying downpayment also.  Files to Modify:   1. EditDownpaymentAmount.jsp   Create a new jsp to provide the user with edit downpaymnet for the selected plan if the flexibleDownpayment is allowed for the selected plan and the project level flag is ON.     1. EditSaleTransaction.jsp/OMEditSaleTransaction.jsp   Modify the above existing jsps to provide an option to modify the downpayment in addition to the existing swap device options.     1. EditSaleTransactions.java   Modify the existing actNextSwapDevice, upgradeNextSwapDevice methods to forward to new jsp when the flexibleDownpayment is allowed for the selected plan and the project level flag is ON. Once the user selected the downpayment amount update the required fields and generate the RIA as BAU. | 288312\_OPUS\_SR\_140  **CR DUE** |

## Functional Overview

|  |  |  |
| --- | --- | --- |
| **Module Name** | **New/Existing** | **Functional description** |
| Activation/ Add a line | Existing | A new NEXT finance plan offer will be available from TLG for basic phones |
| Upgrade Eligibility | Existing | New NEXT finance plan offer will be available from TLG🡪MRE. |
|  |  |  |
|  |  |  |
|  |  |  |

## 

## Middleware Design



### External Interfaces

|  |  |
| --- | --- |
| **Interface ID** | **#1 AppQuery.getUpgradeEligibility** |
| Description | This interface provides upgrade offers for a single CTN. |
| Interface Type | Outbound AND Synchronous |
| Providing/Originating System | TLG API |
| Interface Technology/Protocol | EJB (RMI/IIOP) |
| Service Interface Name | **AppQuery.getUpgradeEligibility** |
| New/Existing | Existing service |
| Change Summary | This API is modified to return EIP offers |
| Throttling Mechanism (for synchronous interfaces) | No change |
| Average response time | ~2 seconds |
| Expected calls per day | 10000 |
| Logging Approach | opus-debug.log |

|  |  |
| --- | --- |
| **Interface ID** | Appquery.getEnrlInstlmntPreValid |
| Description | This interface provides Pre valid Next eligibility. |
| Interface Type | Outbound AND Synchronous |
| Providing/Originating System | TLG API |
| Interface Technology/Protocol | EJB (RMI/IIOP) |
| Service Interface Name | Appquery.getEnrlInstlmntPreValid |
| New/Existing | Existing service |
| Change Summary | This API is modified to return EIP offers |
| Throttling Mechanism (for synchronous interfaces) | No change |
| Average response time | ~2 seconds |
| Expected calls per day | 10000 |
| Logging Approach | opus-debug.log |

### 

### Performance Design

N/A. No new calls are designed in the critical flow path.

### Security Design

### No new SPI elements added/modified as part of this project.

### UI Design

### UI Design – Wireframe

<http://clddev0srv04830:8600/OPUS/OPUS_Launcher_DEV/pdfs/OPUS/OPUSMobile_288312_Next2FlexibleDownPayments.pdf>

### UI Action Processing - Sequence Diagram – No new UI sequence is being designed.

### Asynchronous/Batch Processing - Sequence Diagram

*NA*

### Externalized Configuration

*1 - Store Configurables : These parameters are used for controlling on/off behavior for a unit of functionality(or feature) at various levels (global/channel/sub-channel/store).*

|  |  |  |
| --- | --- | --- |
| **Store Configurables (Repeat for each parameter)** | | |
| *#* | **Parameter name** |  |
|  | **Description** | *OPUS - This parameter will provide individual stores the flexibility to control whether the CARE Credit referral requirements will be implemented on not.* |
|  | **Specification** | 1. *STORE\_CONFIGURABLES\_LIST*   SUPPORT\_NEWNEXT\_FLEXIBLEDP\_PLANS |

*2 - Application Configurables : These parameters cover wide range of externalized configuration needs that are NOT related to switching functionality ON/OFF at channel/sub-channel/store levels (which is already covered by Store Configurables described above).*

|  |  |  |
| --- | --- | --- |
| **Application Configurables (Repeat for each parameter)** | | |
| **#** | **Parameter name** |  |
|  | **Description** | *OPUS/OM - This parameter will be used to configure disclosure message in enterprise\_config table.* |
|  | **Specification** | 1. *CSI\_CONFIG*   *Not Applicable*   1. *ENTERPRISE\_CONFIG - This table contains general configurations (name/value pairs) that don’t fall in any of the categories above.*  |  |  |  |  | | --- | --- | --- | --- | | **VP\_KEY\_NAME** | **CV\_DESCRIPTION** | **EC\_FRIENDLY\_DESC** | **DEVELOPMENT\_LEAD** | | Flexible\_DP\_INELIGIBLE\_PALNTYPE | EIP | If a value is present then OPUS will not allow flexible DP for that plan type |  | | BUY\_BACK\_CREDIT\_MESSAGE |  |  |  | |  |  |  |  | |  |  |  |  | |

## Database Tier Design

N/A

## Alternative Designs

Please refer to SA document for any alternative designs.

## Assumptions

1. Calculation and collection of sales tax for devices sold on new % based NEXT shall leverage existing rules.
2. Buyback credit cannot be used for settling the DF cart within OPUS. The buyback credit is added to the C&C cart and can be used towards settling the amount due in the C&C cart only.

Gift cards are not allowed for DF settlement either. So, if part or whole of the buyback credit is issued on a promo card which is mailed to the customer and the customer walks into the store with that gift card, he cannot use that to make payment for DF items.

1. No additional reporting changes are needed to report on how many buyback credits were applied to down payments. Existing reports already provide information on how buyback credits get used and can be leveraged if needed.
2. TLG Memos are not being enhanced for % based Next offers.
3. OPUS currently supports BTM functionality for financed devices if enabled by Customers contract type. No changes are expected to this functionality. Also, Payup/Payoff BTM is not supported within OPUS (BR-ATTNEXT.030)
4. OPUS is not expected to make any changes in order to reflect the % model information on the RIA and CSS. EDP will handle the changes (BR-ATTNEXT.026).
5. Current process will continue to block pay-up when the customer is not eligible for pay up (X months after start of a new contract)
6. The payup amount will be returned by TLG/MRE in dollars to OPUS.
7. The current reporting process shall be leveraged to allow reporting and accounting to determine what tenders were applied to specific charges in the Cash and Carry shopping cart. No OPUS changes are expected for the same.
8. Legacy plan definition has been clarified as NEXT plan with a Trade-In Term defined, such as NXT12, NXT18. Installment Plan and Smart Device plans are not considered as legacy plans, the new basic phone NEXT plan is not legacy plan either because it does not have Trade-In Term even though it is a NE plan.
9. The new NEXT %model plan functionality in TLG and MRE will be set to Y at pilot for all markets. i.e., in a pilot store, MRE and TLG will send both Legacy and New Next offers.
10. At launch, all OPUS stores will turn ON the new NEXT offers support for a particular channel.

## Constraints

### None

## Pre-Production Disaster Recovery Planning

### NA

## Other Plans and References

**SR link:**

<http://p8prism.web.att.com:80/PRISMP8/EmailLinkServlet?objectID=1C9C6BFF-6337-4200-A39A-5EFE870AA54F&folderid=288312&documentumToP8=NO&moduleid=5&environment=10>

## Acceptance & Approvals

Overview

Use this section to capture approvals in the event that electronic approvals via the PRISM Project Workflow Module will not be used.

The Approvers of this work product agree that this document is acceptable and complete to the best of their knowledge and will be used by the project team as an official deliverable for the project. It is further agreed that this document can now be baselined and any changes to these sections from this point forward must follow the Managing Change in the Technology Development Unified Process.

Embed evidence of approval in the review table below, or use the PRISM Approval Functionality in the Project Workflow Module Workflow Template View.

Approvers

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTUID and Name** | **Role** | **Group/Application** | **Version Approved, Approval Date and Approval Evidence** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |