UAB
Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

This is the **published version** of the bachelor thesis:

Gispert Cortes, Laura. *Optimitzar la planificació de components (OCP).* Treball de Final de Grau (Universitat Autònoma de Barcelona), 2025 (Grau en Enginyeria de Dades)

This version is available at https://ddd.uab.cat/record/337352

under the terms of the  license.

Optimització de la planificació de components (OCP) amb una pipeline de dades sobre Palantir Foundry

Laura Gispert Cortés

Resum– Aquest treball presenta una *pipeline* de dades sobre Palantir Foundry per a l'optimització de la planificació de components (OCP). A partir de dades sintètiques amb estructura de taules SAP, reconstrueix demanda, *forecast*, inventari i comandes a nivell de material–localització, en deriva mètriques agregades i segmenta cada material segons valor econòmic, estabilitat de la demanda, qualitat del *forecast* i variabilitat del termini d'aprovisionament. Segons el segment s'assigna una política d'inventari i, mitjançant simulació de Monte Carlo, s'estimen els paràmetres òptims (punt de comanda, estoc de seguretat i nivell màxim). Un *backtesting* valida les propostes contra l'històric i les compara amb la configuració actual; els resultats es mostren en un *dashboard* fet amb Slate. Es demostra així que és viable convertir dades transaccionals brutes en paràmetres d'inventari accionables i traçables de manera reproduïble.

Paraules clau– optimització d'inventari, Palantir Foundry, dades sintètiques SAP, EOQ, segmentació ABC/XYZ, Monte Carlo, *backtesting*.

Abstract– This work presents a data pipeline on Palantir Foundry for Optimize Component Planning (OCP). Starting from synthetic data with SAP table structure, it rebuilds demand, forecast, inventory and orders at the material–location level, derives aggregated metrics and segments each material by economic value, demand stability, forecast quality and lead-time variability. A policy is assigned per segment and, through Monte Carlo simulation, the optimal parameters (reorder point, safety stock and order-up-to level) are estimated. A backtesting stage validates the proposals against history and compares them with the current setup; results are shown in a Slate dashboard. We show it is feasible to turn raw transactional data into actionable, traceable inventory parameters in a reproducible way.

Keywords– inventory optimization, Palantir Foundry, synthetic SAP data, EOQ, ABC/XYZ segmentation, Monte Carlo, backtesting.

1 Introducció - Context del treball

La gestió d'inventaris és un dels problemes clàssics de l'enginyeria de l'organització industrial i, alhora, un dels que té un impacte econòmic més directe. Mantenir massa estoc immobilitza capital, ocupa espai i exposa el material a obsolescència o a la pèrdua per caducitat; mantenir-ne massa poc provoca trencaments d'estoc, comandes urgents i pèrdua de servei. El repte és trobar, per a cada material, el punt d'equilibri entre el cost de mantenir inventari i el cost de no tenir-ne.

En entorns industrials reals aquesta decisió es complica per tres factors. Primer, el nombre de materials és molt elevat i heterogeni: conviuen components de consum estable amb peces de demanda esporàdica, i cadascun pot comportar-se de manera diferent a cada planta o ubicació. Segon, la informació de partida resideix en sistemes de planificació de recursos empresarials (ERP) com SAP[1], on les dades estan disperses en moltes taules transaccionals (moviments de material, comandes, mestres de material) no pensades directament per a l'anàlisi i sovint amb redundàncies o inconsistències. Tercer, la mateixa política d'inventari adequada per a un material no ho és per a un altre, de manera que no n'hi ha prou amb una fórmula única; cal classificar el catàleg i tractar cada grup amb la lògica que li correspon.

Aquest treball aborda el problema mitjançant una solució de dades extrem a extrem que hem anomenat OCP (*Optimize Component Planning*). El sistema parteix de dades amb estructura SAP, les transforma en un model analític net i, sobre aquest model, calcula de manera automàtica i traçable els paràmetres d'inventari recomanats per a cada material, presentant-los finalment en un *dashboard* de suport a la decisió. La hipòtesi de treball és que, amb la combinació adequada de neteja de dades, segmentació i simulació, es pot passar de dades brutes a recomanacions operativament viables sense intervenció manual material a material.

La resta del document s'organitza de la manera següent. La secció 2 fixa la motivació i els objectius. La secció 3 descriu la planificació i la metodologia. La secció 4 presenta l'entorn de desenvolupament, Palantir Foundry[2]. La secció 5 explica com s'han generat les dades sintètiques basades en SAP. La secció 6 detalla la *pipe-*

• E-mail de contacte: laura.gispert@autonoma.cat

• Treball tutoritzat per: Eduardo Cesar Galobardes (Departament d'Arquitectura de Computadors i Sistemes Operatius)

• Curs 2025/26

[1]SAP és un dels principals sistemes ERP del mercat: un programari de gestió integrada que centralitza els processos d'una empresa (compres, inventari, producció, finances, etc.).

[2]Plataforma d'integració, transformació i operació de dades de Palantir Technologies, que permet emmagatzemar, transformar i explotar dades en un mateix entorn.

line de dades, que és el nucli tècnic del treball. La secció 7 descriu el *dashboard* de visualització construït amb Slate[3]. Finalment, la secció 8 recull els resultats, la discussió i les conclusions, juntament amb les línies de continuació.

2 MOTIVACIÓ I OBJECTIUS

La motivació del treball neix de la distància que sovint existeix entre les dades operatives d'una empresa i les decisions de planificació. Les dades hi són cada moviment de material queda registrat al sistema ERP, però no estan en un format que permeti decidir directament quant i quan demanar de cada component. Sovint aquesta decisió es pren amb regles heretades, valors fixats fa temps o criteris poc homogenis entre planificadors. Aquest treball vol cobrir aquesta distància construint un sistema que converteixi dades transaccionals en paràmetres d'inventari accionables, objectius i reproduïbles.

L'objectiu general és **dissenyar i implementar una *pipeline* de dades reproduïble que, a partir d'estructures SAP, calculi paràmetres d'inventari optimitzats per material i en faciliti la interpretació mitjançant un *dashboard***.

Aquest objectiu general es desplega en els objectius específics següents:

- Generar un conjunt de dades sintètiques que reprodueixi de manera fidel l'estructura de les taules SAP rellevants, per disposar d'un banc de proves realista sense dades confidencials.
- Reconstruir, a partir d'aquestes dades, les magnituds de negoci fonamentals (demanda, *forecast*, inventari i comandes) en un model analític net amb claus tècniques consistents.
- Segmentar els materials combinant criteris econòmics i de comportament de la demanda per assignar a cadascun la política d'inventari més adequada.
- Estimar els paràmetres òptims de cada política mitjançant simulació, equilibrant cost i nivell de servei.
- Validar les propostes amb *backtesting* històric i comparar-les amb la configuració actual.
- Visualitzar els resultats en un *dashboard* interactiu orientat a la presa de decisions.

L'abast del treball se centra en el disseny i la implementació de la *pipeline* i de la seva visualització, i en la validació metodològica sobre dades sintètiques; queda fora de l'abast el desplegament en producció sobre dades reals d'una organització concreta, que es planteja com a línia de continuació.

3 PLANIFICACIÓ I METODOLOGIA

El treball s'ha plantejat amb una metodologia incremental i orientada a dades, en què cada capa de transformació es construeix i es valida abans de passar a la següent. Aquest enfocament és coherent amb l'arquitectura de la *pipeline*, que avança en etapes successives des de les dades brutes fins als resultats finals, de manera que un error en una capa primerenca es detecta abans de propagar-se.

[3]Eina de Palantir Foundry per crear aplicacions web interactives connectades directament als conjunts de dades.

La metodologia s'ha estructurat en cinc grans fases, que es corresponen amb les cinc etapes lògiques de la *pipeline*:

- **Ingesta i preparació** (*Input Pipeline*): ingesta de dades amb estructura SAP, normalització, delimitació de l'abast, neteja, càlcul de l'EOQ, mètriques agregades i segmentació.
- **Optimització de paràmetres**: ajust de distribucions, simulació de Monte Carlo[4] i càlcul de punts de comanda, estocs de seguretat i graelles de paràmetres per política.
- **Post-processament**: selecció de paràmetres òptims, corbes de compromís, *backtesting*, comparació contra la configuració actual i agregació d'inventari.
- **Seguiment del valor** (*Value Tracking*): indicadors clau de rendiment (KPI) incrementals per material–localització que quantifiquen la millora.
- **Proves**: tests de consistència d'entrades i de sortides.

Una característica transversal del disseny és la prioritat per la **traçabilitat**: en cada pas es conserven no només els resultats sinó també les variables intermèdies que els expliquen, de manera que qualsevol valor final es pot resseguir fins a les dades originals. Aquesta decisió encareix lleugerament l'emmagatzematge, però fa el sistema auditable i facilita la depuració.

Pel que fa a la planificació temporal, el treball s'ha desenvolupat al llarg del curs amb lliuraments parcials de progrés que han permès validar de forma iterativa l'estat de la *pipeline* i corregir el rumb. Les eines principals han estat Palantir Foundry com a plataforma de dades i PySpark[5] com a llenguatge de transformació, complementades amb Slate per a la capa de visualització.

Una decisió metodològica important ha estat treballar amb **dades sintètiques** en lloc de dades reals. Això permet desenvolupar i compartir el sistema sense restriccions de confidencialitat, alhora que dóna control total sobre les propietats estadístiques del conjunt de proves (es pot forçar, per exemple, que coexisteixin materials de demanda molt estable i molt erràtica). La contrapartida és que els resultats numèrics no representen un cas real concret, sinó la validesa del mètode.

L'estratègia de validació ha estat, per tant, doble. D'una banda, una validació *tècnica*, garantida per les proves de consistència que recorren la *pipeline*: cada capa comprova que les seves entrades i sortides compleixen les condicions esperades (per exemple, que no hi hagi demandes negatives, que els nivells de servei estiguin dins de rang o que els paràmetres respectin les restriccions de vida útil). De l'altra, una validació *funcional*, basada en el *backtesting*, que comprova que les recomanacions tinguin sentit quan es projecten sobre l'històric. Aquesta separació entre validar que el càlcul és correcte i validar que el resultat és útil ha estat clau per avançar amb confiança, ja que permet localitzar si un problema prové d'un error d'implementació o d'una decisió de modelització.

[4]Mètode que estima resultats mitjançant la generació repetida de mostres aleatòries; aquí simula molts escenaris de demanda i de termini per avaluar el risc de trencament d'estoc.

[5]Interfície de Python per a Apache Spark, un motor de processament de dades distribuït i de gran escala.

4 Entorn de desenvolupament: Palantir Foundry

Palantir Foundry és una plataforma d'integració i operació de dades que ofereix, en un mateix entorn, l'emmagatzematge de conjunts de dades, la seva transformació i l'explotació mitjançant aplicacions. En aquest treball Foundry actua com a columna vertebral de tot el sistema: hi resideixen tant les dades sintètiques d'entrada com totes les taules intermèdies i de sortida de la *pipeline*.

El model de treball a Foundry es basa en *datasets* versionats i en transformacions que els connecten formant un graf de dependències. Cada transformació pren un o més *datasets* d'entrada i en produeix un de sortida; quan una entrada canvia, Foundry sap quines sortides cal recalcular. Aquest enfocament aporta dos avantatges clau per al projecte: el **llinatge de dades** (cada taula sap d'on prové i com s'ha calculat) i la **reproductibilitat** (el graf es pot tornar a executar de manera determinista i obtenir el mateix resultat).

Les transformacions s'han implementat en PySpark, que permet expressar operacions de neteja, agregació i unió sobre grans volums de dades de forma distribuïda. L'ús de finestres (*window functions*) per calcular agregats per grup sense perdre el detall, d'agregacions condicionades i d'unions de diferents tipus (*inner*, *left*, *outer*) és recurrent al llarg de tota la *pipeline*, i constitueix el vocabulari bàsic amb què s'expressa la lògica de negoci.

L'organització de les transformacions en repositoris de codi i la seva connexió en un graf permet, a més, separar clarament les responsabilitats de cada etapa i provarles de manera aïllada. Cada *dataset* de sortida és el resultat d'una transformació documentada, i la plataforma permet inspeccionar-ne l'esquema, les estadístiques bàsiques i l'historial de versions. Aquesta organització ha estat clau per gestionar la complexitat d'una *pipeline* amb desenes de taules intermèdies, ja que fa possible localitzar amb rapidesa on s'introdueix una inconsistència i corregir-la sense haver de recalcular tot el flux.

Sobre la capa de dades, Foundry ofereix Slate, una eina per construir aplicacions web interactives directament connectades als *datasets*. Slate s'ha utilitzat per desenvolupar el *dashboard* final (secció 7), que consulta les taules de sortida de la *pipeline* i les presenta de manera comprensible per a l'usuari de negoci, sense que aquest hagi de conèixer l'estructura de dades subjacent. D'aquesta manera, la mateixa plataforma cobreix tot el cicle, des de la dada bruta fins a la visualització, sense necessitat d'exportar dades a eines externes, cosa que reforça la traçabilitat i evita còpies descontrolades de la informació.

5 Generació de dades sintètiques basades en estructures SAP

Per disposar d'un banc de proves realista sense utilitzar dades confidencials, s'han generat dades sintètiques que reprodueixen l'estructura de les taules SAP rellevants per a la planificació. Les taules de referència utilitzades es recullen a l'Apèndix A1; les principals són els moviments de material (MSEG/MKPF), els mestres de material (MARA, MARC, MAKT), la informació de proveïdor (EINA/EINE) i l'inventari per ubicació d'emmagatzematge.

La reconstrucció de la **demanda** s'ha fet a partir dels moviments de material. Es filtren els tipus de moviment (`BWART`) rellevants 201, 261 i 262 i s'interpreta el 262 com a reversa del 261: la seva quantitat es multiplica per -1 i es reetiqueta com a 261. Després s'agreguen els moviments per material, planta, lot, ordre, tipus de moviment i document de vendes, de manera que el consum net no es comptabilitzi dues vegades. Finalment s'uneix amb la capçalera del document per obtenir la data comptable. El resultat és una taula de demanda amb les columnes `location_id`, `material_id`, `period` i `requested_qty`.

El **forecast** mereix una nota important. A banda de normalitzar un *forecast* extern, el sistema genera un *forecast* sintètic a partir de la demanda real multiplicant-la per un factor aleatori estable entre 0,4 i 1,6, fixat amb una llavor per material. Així, cada material té un patró d'error reproduïble. Cal subratllar que aquest *forecast* és una **generació sintètica** amb finalitat de prova, no una predicció real; el *forecast* final és la unió del component extern normalitzat i d'aquest component sintètic.

L'**inventari** es transforma d'un format ample (una columna per tipus d'estoc) a un format llarg, amb una fila per tipus: *unrestricted*, *intransit*, *restricted*, *quality* i *blocked*. Aquí cal documentar una particularitat: l'estoc restringit se suma a l'*unrestricted* alhora que es manté també com a fila pròpia, cosa que pot provocar doble comptabilització si en una anàlisi agregada se sumen tots els tipus sense correcció.

El **mestre de material** integra MARA, MARC i MAKT, i assigna el controlador MRP (`DISPO`) dins d'un sistema SAP, l'MRP (*Material Requirements Planning*) és el procés de planificació de necessitats de material que determina quan i quant cal reaprovisionar de cada material, i el controlador és el responsable assignat a cada material per gestionar-lo mitjançant una simulació categòrica: amb una llavor fixa es fa un creuament contra una distribució acumulada i s'escull el primer valor que la supera, cosa que garanteix que l'assignació, tot i ser aleatòria, sigui reproduïble. L'ús de llavors fixes en tots els components aleatoris (*forecast* sintètic, assignació de `DISPO`) és una decisió de disseny important: assegura que dues execucions de la *pipeline* sobre les mateixes dades produeixin exactament el mateix resultat, condició necessària per poder depurar el sistema i comparar versions de manera fiable. Finalment, la informació de **proveïdor** s'obté unint EINA i EINE per relacionar proveïdor, material i planta; es filtren els registres no esborrats, es normalitza el preu i s'agreguen dades de proveïdor mitjançant finestres per material–localització.

Les **comandes** i els **terminis d'aprovisionament** completen el conjunt de dades. El termini real es reconstrueix a partir de les dates de comanda i de lliurament registrades, i s'agrega per material–localització per obtenir-ne la mitjana i la desviació estàndard. Aquestes dues magnituds són especialment importants, perquè la variabilitat del termini i no només la seva durada condiciona l'estoc de seguretat necessari: un material amb un termini llarg però molt estable pot requerir menys protecció que un material amb un termini curt però imprevisible. Per això el sistema no es limita a fer servir un termini mitjà, sinó que en conserva la dispersió per traslladar-la a la simulació posterior. El resultat de tota aquesta fase és un conjunt de taules normalitzades demanda, *forecast*, inventari, comandes, terminis, mestre de material i proveïdor amb una estructura consistent i preparades per ser

delimitades i netejades en la fase següent.

6 DESENVOLUPAMENT DE LA PIPELINE DE DADES

La *pipeline* és el nucli del treball. La unitat analítica central durant gairebé tot el flux és el parell **material–localització**, identificat per la clau `material_location_id`. Gairebé totes les taules acaben generant aquesta clau, i les taules d'esdeveniments hi afegeixen el període, la comanda o el tipus d'inventari per construir claus úniques. Aquesta tria de la unitat d'anàlisi és deliberada: el mateix material a dues plantes diferents per exemple, el material A a la planta 1 i el material A a la planta 2 es tracta com a dues unitats independents, cadascuna identificada per la seva clau `material_location_id`, perquè poden tenir demanda, terminis d'aprovisionament i fins i tot polítiques diferents. Optimitzar per material sense distingir la localització portaria a recomanacions poc realistes. Per això tot el flux des de la demanda fins als paràmetres finals es manté desagregat a aquest nivell, i només s'agrega quan cal oferir una visió de conjunt al *dashboard*.

El flux avança per capes successives: primer es preparen i delimiten les dades d'entrada; després es calcula l'EOQ i les mètriques agregades; tot seguit es segmenta cada material i se li assigna una política; a continuació s'optimitzen els paràmetres mitjançant simulació; i finalment es valida la proposta amb *backtesting* i s'agreguen els resultats. A continuació es descriuen aquestes etapes principals.

6.1 Preparació i delimitació de l'abast

Després de la ingesta, una fase d'*scoping* restringeix les dades als materials i localitzacions vàlids per a l'optimització. Les regles d'abast estan parametritzades (primera lletra del material, tipus de material, controlador MRP, estat del material, tipus d'aprovisionament, etc.) i s'apliquen per localització: cada paràmetre pot contenir una llista de valors separats per comes, i un valor nul es tradueix en `NO_CRITERIA`, és a dir, no aplicar el filtre. La regla general és que el mestre material–localització mana: qualsevol transacció que no hi tingui correspondència s'elimina.

A continuació, una fase de neteja final aplica validacions de consistència (costos, nivells de servei, demanda, materials, terminis, *forecast* i segmentació) i genera les claus tècniques del model:

- `demand_id` = material–localització–període.
- `inventory_id` = material–localització–període–tipus d'inventari.
- `order_id` = combinació de material, localitzacions origen/destí, número i posició de comanda i data de lliurament.
- `material_location_id` = material–localització.

Així, la *pipeline* no només neteja dades sinó que prepara un model relacional consistent per a les etapes posteriors.

L'objectiu d'aquesta fase és doble. D'una banda, reduir el volum de dades: el conjunt inicial conté molts més materials i moviments dels que són rellevants per a l'optimització, i el filtratge permet centrar el càlcul en allò que realment importa. De l'altra, i més important de cara a un ús real, recollir les condicions de negoci de cada organització. A la pràctica, cada empresa té els seus criteris i prioritats quins materials gestiona activament, quins exclou i quines restriccions ha de tenir en compte i és precisament aquesta lògica la que el *scope* permet codificar de manera parametritzada, sense modificar el codi. Saber com treballa cada organització i a què dóna prioritat és essencial perquè el resultat sigui realista, ja que sovint cal deixar fora certs materials o aplicar restriccions específiques que un model genèric no copsaria.

6.2 Càlcul de l'EOQ

Per a cada material–localització es calcula la quantitat econòmica de comanda (EOQ) a partir de la fórmula clàssica de Harris–Wilson

$$Q^* = \sqrt{\frac{2\,D\,C_o}{C_h}} \tag{1}$$

on D és la demanda anual, C_o el cost de llançar una comanda i C_h el cost de mantenir una unitat en estoc durant un any. La demanda anual a partir d'històric es calcula com la mitjana de les demandes anuals observades (suavitzant diferències entre anys), mentre que la variant basada en *forecast* pren la suma dels primers dotze mesos disponibles. El cost de mantenir es deriva del cost mensual anualitzat ($\times 12$), i el cost de comanda es reparteix entre els materials proporcionalment al seu temps d'inspecció, només per als rellevants per a control de qualitat.

El valor teòric de l'equació (1) es corregeix després amb restriccions operatives encadenades: límit per la demanda anual, quantitat mínima de comanda (MOQ), arrodoniment al múltiple de lot fix, límit per vida útil

$$Q_{\text{shelf}} = \frac{D}{365} \cdot T_{\text{shelf}} \tag{2}$$

i, finalment, quantitat màxima de comanda. Aquestes restriccions transformen un òptim purament matemàtic en un valor operativament viable.

El càlcul es resol en tres passos. Un primer script obté l'EOQ a partir de la demanda històrica; un segon el calcula a partir del *forecast*; i un tercer els consolida, conservant els materials que només tenen un dels dos càlculs. Disposar de les dues variants històric i previsió permet triar la més adequada segons si el material es governa per consum o per *forecast*.

6.3 Mètriques agregades

Sobre les dades netes es calculen mètriques agregades que consoliden demanda, *forecast*, inventari i comandes i que alimenten la segmentació i les restriccions de paràmetres. Entre les principals hi ha la demanda anual i mitjana, l'error de *forecast*, les comandes i l'inventari agregats i la demanda mòbil (*rolling*). El coeficient de variació de la demanda, clau per a la classificació posterior, es defineix com

$$CV = \frac{\sigma_D}{\mu_D} \cdot 100 \tag{3}$$

on μ_D i σ_D són la mitjana i la desviació estàndard de la demanda. La precisió del *forecast* es mesura per períodes (mensual i trimestral) com a complementari de l'error relatiu, i el biaix com el percentatge de períodes en

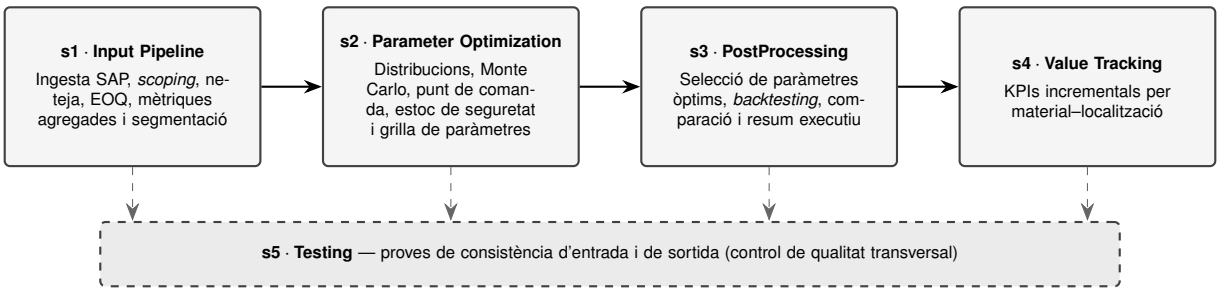

Fig. 1: Arquitectura de la *pipeline* OCP: les quatre etapes seqüencials de transformació (s1–s4), des de la ingesta de dades SAP fins al seguiment de valor, i la capa transversal de proves de consistència (s5) que actua com a control de qualitat abans i després de l'optimització.

què el *forecast* supera la demanda. La finestra temporal emprada per a la demanda i el *forecast* històrics és, aproximadament, els darrers 730 dies fins al darrer dia tancat del mes anterior. Una senyal addicional és la freqüència de demanda (setmanal/mensual), que permet distingir materials de consum freqüent de materials de demanda baixa o intermitent.

A banda del coeficient de variació, en aquesta capa es consoliden també el termini d'aprovisionament mitjà (*Order_LT*), la seva desviació estàndard com a mesura de variabilitat, el nombre de comandes, la demanda total, mitjana i acumulada, i les mètriques equivalents per al *forecast*. Aquestes magnituds són les que, combinades, alimenten tant la segmentació com les restriccions que s'apliquen més endavant en l'optimització de paràmetres. La importància d'aquesta capa rau en que homogeneïtza el càlcul: totes les etapes posteriors consumeixen les mateixes mètriques, calculades una sola vegada i sobre la mateixa finestra temporal, cosa que evita discrepàncies entre criteris i facilita la comparació entre materials.

6.4 Reconstrucció de terminis i senyals de demanda

El termini d'aprovisionament real es reconstrueix a partir de les dates de comanda i de lliurament registrades a les ordres, i s'agrega per material–localització per obtenir-ne tant la mitjana com la desviació estàndard. Aquesta darrera és la que mesura la variabilitat del termini, una de les dimensions clau de la segmentació: un termini llarg però estable és més fàcil de cobrir que un de curt però imprevisible, perquè permet planificar la reposició amb confiança.

A més del termini, es calcula una senyal de freqüència de demanda que classifica cada material segons si té consum setmanal, mensual o més espaiat. Aquesta senyal permet separar els materials de consum freqüent per als quals les mètriques de variabilitat són estadísticament fiables dels materials de demanda baixa o intermitent, on un coeficient de variació alt no indica necessàriament inestabilitat sinó simplement escassetat de dades. Distingir aquests dos casos evita penalitzar amb estocs de seguretat elevats materials que només es demanen ocasionalment, i és el que justifica l'existència de segments específics per a la demanda baixa o irregular.

6.5 Segmentació i assignació de política

La segmentació és el cor funcional de la *pipeline*. Combina diverses dimensions. Per valor econòmic s'aplica una classificació **ABC** de Pareto sobre el volum

$$V = \text{valoració} \times \text{demanda anual} \tag{4}$$

assignant A fins al 80% del volum acumulat, B fins al 95% i C la resta. Per variabilitat de la demanda s'aplica una classificació **XYZ** segons el coeficient de variació (X si CV < 35, Y si $35 <$ CV ≤ 80, Z si CV > 80). A més, es distingeix entre materials **governats per *forecast*** (segments SF) i **governats per consum** (segments SC).

La classificació de *forecast* és especialment detallada. Per als materials amb *forecast* actiu s'avaluen la precisió mensual i trimestral i el biaix; només si superen els llindars de precisió (≥ 50) i de biaix (< 80) es mantenen com a governats per *forecast*, i altrament es reclassifiquen a consum. Dins dels governats per consum, la combinació de coeficient de variació de la demanda i de variabilitat del termini distingeix entre materials estables (SC1), estables amb termini variable (SC1b), de variabilitat mitjana (SC2/SC2b) i erràtics (SC3), amb un segment de reserva (SC4) per als casos incoherents. La Taula 1 resumeix la lògica de cada segment.

TAULA 1: SEGMENTS I CRITERIS PRINCIPALS

Segment	Criteri principal
SF1	*Forecast* fiable, CV alt (≥ 80)
SF1b	*Forecast* fiable, CV baix (< 80)
SF1c	LT curt, precisió ≥ 75, SS$= 0$, VERP
SC1	CV ≤ 35, termini estable
SC1b	CV ≤ 35, termini variable
SC2	$35 <$ CV ≤ 80, termini estable
SC2b	$35 <$ CV ≤ 80, termini variable
SC3	CV > 80 (demanda erràtica)
SC4	Cas incoherent (reserva)

A partir del segment s'assigna una **política d'inventari proposada** (per exemple Ss min-max, FMs amb revisió periòdica, Qs o FQs amb quantitat fixa) i un nivell de servei objectiu, mitjançant taules mestres de polítiques. El resultat automàtic encara pot ser ajustat per regles de negoci pròpies de cada organització, però la lògica general és aquesta: una combinació de criteris que permet que la política s'adapti al comportament real de cada material en lloc d'aplicar una fórmula única.

6.6 Definició de les polítiques d'inventari

Les polítiques que la *pipeline* pot assignar responen a quatre famílies, cadascuna adequada a un patró de comportament diferent. La política **Ss** (min-max) defineix un nivell mínim s el punt de comanda i un nivell màxim S:

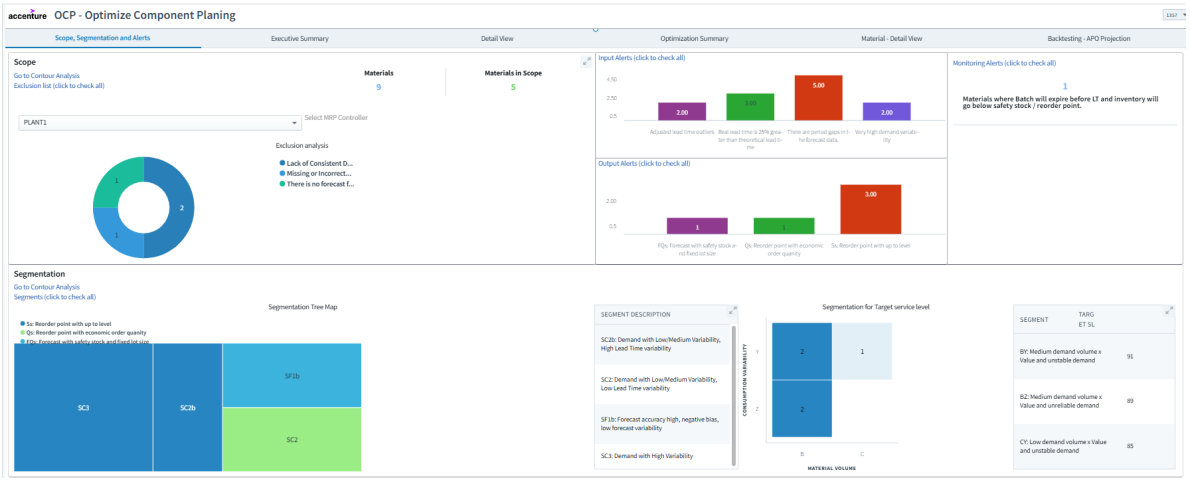

Fig. 2: Vista d'abast i segmentació del *dashboard*: distribució del catàleg en àmbit, mapa d'arbre de segments, classificació per nivell de servei objectiu i alertes d'entrada i sortida generades per la *pipeline*.

quan la posició d'estoc baixa de s es llança una comanda que la porta fins a S. És adequada per a materials de consum estable, on la demanda és prou regular per fixar tots dos llindars. La política **FMs** (*forecast* amb revisió periòdica i estoc de seguretat) revisa l'estoc cada R períodes i demana fins a un nivell objectiu calculat a partir del *forecast*; s'aplica a materials governats per previsió, on té sentit anticipar les necessitats futures en lloc de reaccionar al consum passat.

Les dues famílies restants es basen en una quantitat de comanda fixa. La política **Qs** (consum amb quantitat fixa) demana una quantitat fixa típicament l'EOQ cada vegada que es travessa el punt de comanda, dimensionant les necessitats amb la demanda històrica; és apropiada per a materials de consum amb certa variabilitat. La política **FQs** (*forecast* amb quantitat fixa) segueix el mateix esquema de quantitat fixa, però estima les necessitats a partir del *forecast* en lloc del consum. L'assignació d'una d'aquestes polítiques a cada material es fa segons el seu segment mitjançant taules mestres, i pot ser ajustada per regles de negoci pròpies de cada organització.

6.7 Optimització de paràmetres per Monte Carlo

Per a cada material s'ajusten fins a cinc distribucions estadístiques: la de la demanda, la de l'error de *forecast*, la del termini d'aprovisionament, la de la demanda durant el termini i la de l'error de *forecast* durant el termini. Aquesta darrera parella és especialment important, perquè el risc de trencament d'estoc no depèn de la demanda d'un període qualsevol sinó de la que s'acumula precisament durant el temps que es triga a reposar; modelar conjuntament demanda i termini permet capturar com es combinen les dues fonts d'incertesa. A partir d'aquestes distribucions, una **simulació de Monte Carlo** genera mostres amb què s'avaluen diferents valors de punt de comanda i estoc de seguretat en termes de cost i de nivell de servei. El criteri de selecció depèn de l'objectiu: si el nivell de servei objectiu és baix (≤ 50), s'escull l'òptim per cost; si és alt (> 50), s'escull per assolir el nivell de servei. El punt de comanda resultant respon, conceptualment, a

$$RoP = \bar{D}_{LT} + SS, \qquad SS = z \cdot \sigma_{DLT} \tag{5}$$

on $\bar{D}_{LT}$ és la demanda mitjana durant el termini, z el factor associat al nivell de servei i σ_{DLT} la desviació de la demanda durant el termini; la simulació permet, però, capturar comportaments que la fórmula tancada no recull (per exemple, distribucions asimètriques o demanda intermitent).

L'avantatge de l'enfocament de Monte Carlo respecte de les fórmules analítiques clàssiques és precisament aquest: en lloc de suposar que la demanda durant el termini segueix una distribució normal, ajusta una distribució a les dades observades de cada material i en genera mostres. Així, materials amb demanda molt esporàdica o amb terminis d'aprovisionament molt variables casos en què la hipòtesi de normalitat falla reben un estoc de seguretat coherent amb el seu risc real. El cost computacional és més alt, però queda justificat per la qualitat de l'estimació en els materials difícils, que són sovint els que més impacte tenen en els trencaments d'estoc.

El procediment de simulació recorre un rang de valors candidats de punt de comanda i d'estoc de seguretat. Per a cada candidat genera un nombre elevat de rèpliques del cicle de reposició i, en cadascuna, comptabilitza si s'hauria produït un trencament d'estoc i quin cost (manteniment, comanda i ruptura) hauria comportat. En agregar les rèpliques s'obté, per a cada candidat, un nivell de servei esperat i un cost esperat; és sobre aquesta graella d'avaluacions que s'aplica el criteri de selecció descrit abans. D'aquesta manera, el valor finalment escollit no prové d'una fórmula tancada sinó d'una exploració explícita del compromís entre cost i servei per a aquell material concret.

A partir d'aquests punts de comanda i estocs de seguretat, juntament amb la segmentació i les dades mestres, es construeixen els paràmetres base de cada material. Finalment, una graella de paràmetres aplica les regles específiques de cada política. Per a la política **Ss** (min-max) es restringeix el nivell màxim S perquè no superi el màxim d'estoc admissible; si no hi ha cap combinació vàlida, s'utilitza aquest màxim com a S i punt de comanda. Per a **FMs** (*forecast* amb revisió periòdica) s'apliquen restriccions per vida útil, tolerància i cobertura mensual, i el període de revisió R es calcula a partir de l'EOQ i la demanda anual, fixant-lo com a mínim al termini d'aprovisionament i com a màxim a dotze mesos. Les polítiques **Qs** i **FQs** (quantitat fixa, sobre consum o *forecast* respectivament) apliquen restriccions anàlogues per vida útil i

tolerància. El resultat és el conjunt de paràmetres definitius per material.

6.8 Post-processament i backtesting

L'última etapa selecciona els paràmetres òptims per material i construeix corbes de compromís cost–servei, que permeten analitzar com varia l'inventari i el cost en funció del nivell de servei objectiu i triar el punt d'equilibri desitjat. El component central, però, és el **backtesting**, que es resol en tres passos per a cadascuna de les quatre famílies de política (FMs, FQs, Qs/sQ i Ss). El primer pas *calcula les comandes* que s'haurien generat període a període aplicant la lògica de la política als paràmetres proposats: en una política de punt de comanda, per exemple, es genera una comanda cada vegada que la posició d'estoc baixa del punt de comanda; en una de revisió periòdica, es revisa cada R períodes i es demana fins al nivell objectiu. El segon pas *simula l'evolució de l'inventari* resultant d'aquestes comandes, tenint en compte el termini d'aprovisionament i la demanda real de cada període, de manera que es reprodueix què hauria passat realment amb l'estoc. El tercer pas *calcula els KPIs* (cost total, nivell de servei assolit, trencaments d'estoc, *backlog* i inventari mitjà) a partir d'aquesta simulació. Els mateixos KPIs es calculen per a la configuració actual, de manera que es pot comparar directament la proposta OCP amb la situació de partida sobre una base homogènia.

Les quatre famílies comparteixen aquest esquema de tres passos, però es diferencien en com generen les comandes simulades. En les polítiques de quantitat fixa (Qs i FQs), cada comanda té la mida del lot calculat i es dispara en travessar el punt de comanda. En la política de revisió periòdica (FMs), les comandes només es poden generar en els instants de revisió, i la seva mida s'ajusta per assolir el nivell objectiu fixat a partir del *forecast*. En la política min-max (Ss), la comanda porta l'estoc fins al nivell màxim S cada vegada que es travessa el mínim. Aquestes diferències fan que el perfil d'inventari resultant i, per tant, els KPIs sigui característic de cada política, cosa que permet comparar-les de manera justa sobre el mateix històric i triar la que ofereix el millor compromís per a cada material.

Les corbes de compromís cost–servei complementen aquesta comparació: en lloc d'oferir un únic valor, mostren com evolucionen l'inventari mitjà i el cost total a mesura que s'exigeix més nivell de servei. Cada material té la seva corba, i el seu pendent indica com de car és, en termes d'estoc, guanyar servei addicional; els materials erràtics o de termini molt variable presenten corbes molt més inclinades que els estables. Aquesta representació és la que permet decidir de manera informada el nivell de servei objectiu, especialment quan cal arbitrar entre l'objectiu de negoci i el pressupost d'inventari disponible.

Les corbes de compromís mereixen una atenció especial, ja que tradueixen el problema d'inventari en una decisió de negoci explícita: per a cada material relacionen el nivell de servei amb l'inventari mitjà i el cost associat, de manera que es pot veure quant costa cada punt addicional de servei. Aquesta informació és la que permet justificar per què un material de valor alt i demanda estable pot assolir un servei elevat amb poc estoc, mentre que un material erràtic requereix molt més estoc per al mateix servei. La comparació contra la **configuració actual** tanca el cicle: els mateixos indicadors es calculen amb els paràmetres vigents, de manera que la millora no s'expressa en termes absoluts sinó com a diferència respecte del punt de partida real.

A continuació, una capa de **seguiment del valor** calcula KPIs incrementals per material–localització, que quantifiquen la millora aportada per la proposta i permeten prioritzar on actuar primer. Una capa final agrega l'inventari a diferents nivells per ubicació i període, per ubicació, per tipus de material i per material per alimentar el *dashboard*. Aquesta agregació multinivell és el que permet que una mateixa dada serveixi tant per al resum executiu, que necessita totals per planta o per tipus, com per a la vista de detall, que treballa material a material. Finalment, un conjunt de proves de sortida verifica que els paràmetres compleixen les restriccions de cobertura mensual, vida útil i nivell de servei abans de donar-los per vàlids. Aquestes proves, juntament amb les d'entrada, actuen com a control de qualitat entre les dades sintètiques i la lògica OCP, i són les que garanteixen que cap paràmetre matemàticament òptim però operativament inviable arribi a la sortida.

6.9 Validació de consistència

La darrera peça de la *pipeline* és una capa de proves automàtiques que actua com a control de qualitat entre les dades sintètiques i la lògica OCP. Es divideix en dos blocs. Les **proves d'entrada** s'executen abans de l'optimització i comproven que les dades de partida són coherents: validen el *forecast*, la segmentació, els costos, els nivells de servei, la demanda, els materials, els terminis d'aprovisionament i els reemplaçaments. Per exemple, verifiquen que no hi hagi costos negatius, que els nivells de servei objectiu estiguin dins de rang o que cada material assignat a un segment compleixi les condicions d'aquell segment. Si una d'aquestes condicions falla, el problema es detecta a l'origen i no es propaga cap a les etapes de càlcul.

Les **proves de sortida** s'executen sobre els paràmetres ja calculats i comproven que siguin operativament admissibles: validen la cobertura en mesos de l'estoc de seguretat, el respecte de la vida útil, els nivells de servei assolits, la coherència dels materials optimitzats, el nivell màxim (*up-to level*) i l'inventari resultant. Aquesta doble barrera abans i després de l'optimització és el que permet afirmar que cap paràmetre matemàticament òptim però operativament inviable arriba a la sortida, i és també el que fa que la *pipeline* sigui robusta davant de canvis en les dades d'entrada: si un dia les dades deixen de complir una hipòtesi, les proves ho assenyalen en lloc de produir silenciosament un resultat incorrecte.

7 Dashboard i visualització dels resultats (Slate)

Els resultats de la *pipeline* es presenten en un *dashboard* interactiu desenvolupat amb Slate, l'eina d'aplicacions de Palantir Foundry. L'objectiu del *dashboard* és fer accionable tota la cadena de càlcul anterior, oferint diverses perspectives complementàries que van del global al detall.

La primera és el **resum executiu**, pensat per a una visió global. Mostra els costos d'inventari desglossats (manteniment, comanda i *backlog*), el valor de l'inventari

i el resum d'optimització, és a dir, l'efecte agregat de passar de la configuració actual a la proposta OCP. Aquesta vista permet quantificar d'un cop d'ull el benefici potencial de l'optimització (vegeu l'Apèndix A3).

La segona és la **vista de detall per material**, que permet baixar al nivell d'un material–localització concret. Per a cada material es mostren l'evolució de la demanda enfront del *forecast*, la comparativa de la projecció d'inventari amb els paràmetres actuals enfront dels optimitzats (*baseline vs optimized*), l'evolució del valor d'inventari i un conjunt d'alertes de monitoratge. Aquesta vista és l'eina principal de diagnòstic: connecta el resultat numèric amb el comportament real del material i en justifica la política assignada (vegeu l'Apèndix A4).

El *dashboard* també incorpora una vista d'**abast i segmentació**, que mostra com s'han classificat els materials i quines alertes s'han generat, donant transparència a les decisions preses per la *pipeline*, i una vista de **backtesting** que projecta l'evolució de l'inventari sota els paràmetres proposats. El conjunt de captures principals es recull als apèndixs A2–A4.

Un aspecte clau del *dashboard* és la interactivitat: l'usuari pot filtrar per localització, per segment o per material concret, i les visualitzacions es recalculen en conseqüència. Això permet recórrer el sistema de dalt a baix del resum executiu agregat fins a un únic material seguint un fil narratiu coherent, de manera que el planificador pot començar identificant on hi ha el major potencial de millora i acabar entenent el perquè d'una recomanació concreta. Les alertes integrades (per exemple, sobre dades de *forecast* incompletes, terminis d'aprovisionament anòmals o canvis de paràmetres deguts a restriccions de màxim d'estoc) afegeixen una capa de diagnòstic que ajuda a distingir els casos que requereixen revisió manual dels que es poden acceptar directament. Aquesta combinació de visió agregada, detall per material i alertes és el que converteix el *dashboard* en una eina de suport a la decisió i no en un simple informe estàtic.

8 Resultats, discussió i conclusions

8.1 Resultats

El principal resultat del treball és el propi sistema funcional: una *pipeline* reproduïble que, partint de dades amb estructura SAP, recorre totes les etapes ingesta, *scoping*, EOQ, segmentació, optimització per Monte Carlo i *backtesting* i acaba produint, per a cada material–localització, un conjunt de paràmetres d'inventari recomanats i traçables. El *dashboard* de Slate confirma que aquests resultats es poden comunicar de manera comprensible, tant a nivell agregat (resum executiu) com individual (vista de detall).

Aquest resultat no és menor si es té en compte el punt de partida: desenes de taules SAP transaccionals, amb informació dispersa i redundant, que per si soles no permeten decidir res. La *pipeline* demostra que aquesta matèria primera es pot convertir, de manera automàtica, en un model relacional net i en un conjunt de recomanacions coherents, sense necessitat que un planificador revisi material a material. El nombre de transformacions encadenades ingesta, normalització, *scoping*, neteja, EOQ, mètriques, segmentació, optimització, *backtesting* i agregació és elevat, i el fet que el resultat final segueixi sent traçable fins a l'origen és, en si mateix, un indicador que l'arquitectura escollida és adequada per a un problema d'aquesta complexitat.

Convé remarcar què aporta aquest enfocament respecte de les regles de planificació tradicionals. En molts entorns, els paràmetres d'inventari es fixen amb criteris uniformes o heretats que no distingeixen el comportament real de cada material i que rarament es revisen. El sistema OCP, en canvi, deriva cada paràmetre d'una cadena explícita segmentació, distribucions ajustades i simulació que es pot inspeccionar i justificar pas a pas. El resultat no és només un conjunt de números, sinó una explicació del perquè de cada recomanació: per què un material rep una política de revisió periòdica i un altre una de quantitat fixa, o per què dos materials amb la mateixa demanda mitjana reben estocs de seguretat diferents a causa de la seva variabilitat. Aquesta capacitat d'explicació és, en la pràctica, tan valuosa com els valors concrets, perquè és la que permet que els planificadors confiïn en el sistema i el facin servir en lloc de recórrer a la seva experiència.

El **backtesting** aporta el resultat metodològicament més rellevant: en simular sobre l'històric el comportament que haurien tingut els paràmetres proposats, permet comparar de manera objectiva la proposta OCP amb la configuració actual en termes de cost i nivell de servei. Aquesta comparació és la que dóna credibilitat a les recomanacions, ja que no es limita a calcular un òptim teòric sinó que n'estima el comportament que hauria tingut.

A nivell de procés, el treball valida també que la segmentació compleix la seva funció discriminadora: els materials de valor alt i demanda estable (A–X) reben polítiques i nivells de servei diferents dels materials erràtics o de baix valor, cosa que evita aplicar el mateix tractament a tot el catàleg. La combinació de la classificació ABC/XYZ amb la qualitat del *forecast* i la variabilitat del termini permet que cada material rebi una política coherent amb el seu comportament real, que és precisament l'objectiu de la planificació de components.

Finalment, el treball aporta un resultat de caràcter més qualitatiu però igualment rellevant: la viabilitat de cobrir tot el cicle des de la dada SAP bruta fins a la recomanació visualitzada en una sola plataforma i de manera totalment traçable. Cada paràmetre recomanat es pot resseguir fins a les transformacions i les dades que l'han originat, cosa que en un entorn de planificació real seria clau per guanyar la confiança dels planificadors, que difícilment adoptarien recomanacions que no poden entendre ni qüestionar. La transparència del sistema, materialitzada en el *dashboard* i en la conservació de les variables intermèdies, és per tant part del resultat i no només un detall d'implementació.

8.2 Discussió i limitacions

La principal limitació d'aquest treball deriva del fet d'haver-lo desenvolupat en un entorn hipotètic, sobre dades sintètiques i sense contacte amb un client real. Això ha impedit conèixer de primera mà com una organització tracta realment les restriccions de materials i com fixa els llindars i les regles de negoci que el sistema utilitza. Per aquest motiu, aquests llindars i regles s'han hagut d'assumir, i en un cas real podrien ser diferents: només el contacte directe amb l'empresa permetria ajustar-los i preveure de manera més realista el comportament del model. Reconèixer aquesta limitació no resta valor al tre-

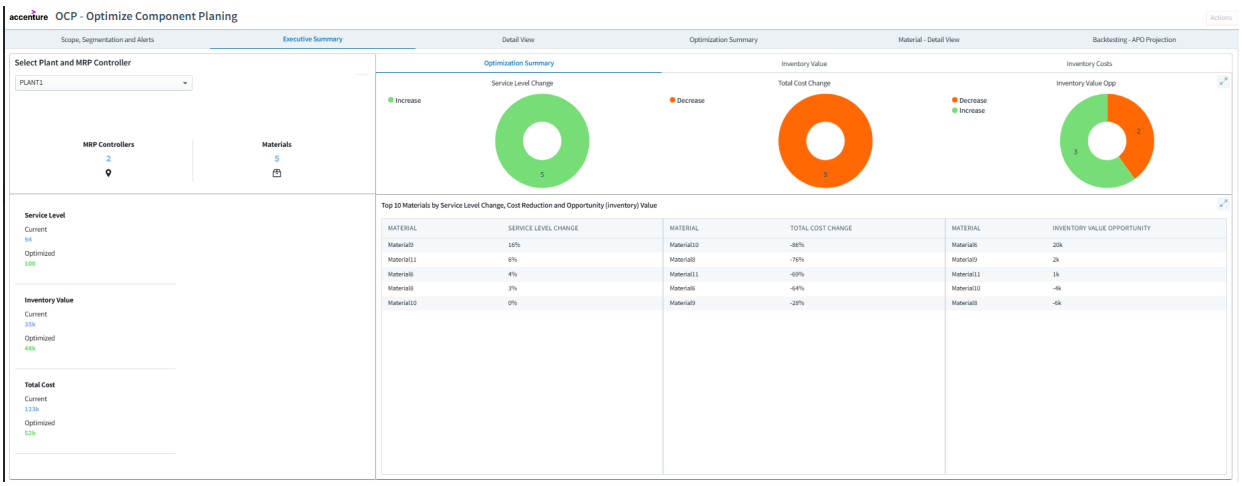

Fig. 3: Resum executiu del *dashboard*: comparació entre la configuració actual i l'optimitzada en nivell de servei, valor d'inventari i cost total, amb el rànquing dels materials amb més variació. Les xifres són sobre dades sintètiques i només il·lustren el comportament relatiu del mètode.

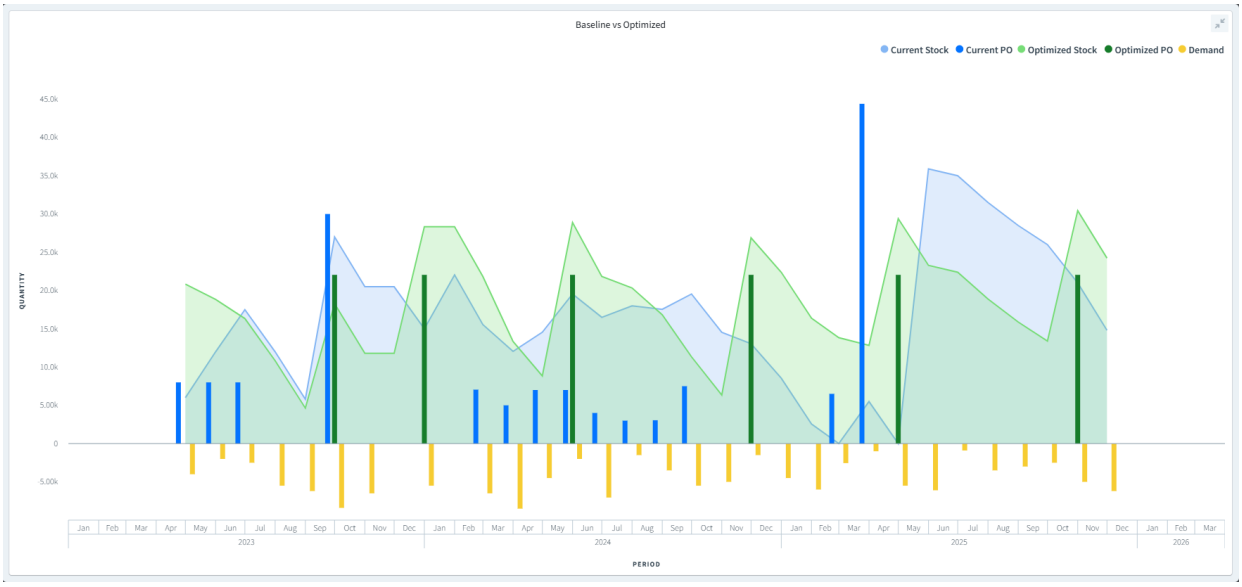

Fig. 4: Projecció comparada de l'estoc al llarg del temps amb la configuració actual (*baseline*) i amb els paràmetres optimitzats, juntament amb les comandes i la demanda de cada període. Il·lustra com l'optimització redueix els nivells d'estoc mantenint la cobertura de la demanda.

ball, sinó que en delimita honestament l'abast.

Cal interpretar amb cura, a més, el significat dels resultats numèrics. Com que les dades són sintètiques, les xifres concretes de cost o de nivell de servei no corresponen a cap escenari real i no s'han de llegir com una estimació d'estalvi. El que sí que és vàlid és el comportament relatiu i el mètode: que la proposta OCP millori els indicadors respecte de la configuració de partida sobre el mateix conjunt de dades demostra que la cadena de càlcul funciona i que el *backtesting* discrimina correctament. La validació definitiva en termes d'estalvi real requeriria, com s'indica a les conclusions, aplicar el sistema sobre dades d'una organització i contrastar les prediccions del *backtesting* amb el comportament observat posteriorment.

Una consideració addicional és la dependència del sistema respecte de taules mestres externes polítiques, nivells de servei objectiu i configuració d'abast que codifiquen el coneixement de negoci. Aquesta dependència és alhora una fortalesa i una limitació: permet que l'organització ajusti el comportament del sistema sense tocar el codi, només mantenint aquestes taules, però també implica que reproduir exactament els resultats en un altre entorn requereix disposar d'aquestes mateixes taules o d'equivalents. En un desplegament real, doncs, una part important de l'esforç no seria estrictament tècnica sinó de governança: definir i mantenir aquestes taules de configuració amb criteris consistents, i acordar qui n'és responsable. Aquesta observació no és menor, perquè explica per què projectes d'aquest tipus fracassen sovint no pel model sinó per la qualitat i el manteniment de la configuració que l'envolta.

8.3 Conclusions i línies de continuació

Aquest treball demostra que és viable construir, sobre Palantir Foundry i a partir d'estructures SAP, una cadena de dades completa que transformi informació transaccional bruta en paràmetres d'inventari accionables, equilibrant cost i nivell de servei i validant les propostes amb *backtes-*

ting. El valor principal no rau en un número concret sinó en el mètode: una segmentació que adapta la política a cada material i una arquitectura traçable i reproduïble de punta a punta.

Com a principal línia de continuació es proposa incorporar una capa d'intel·ligència artificial capaç d'analitzar automàticament els resultats i de respondre preguntes formulades en llenguatge natural sobre qualsevol material o decisió. L'objectiu és que, en el futur, el sistema no només calculi els paràmetres sinó que esdevingui una eina d'assessorament: un assistent al qual el planificador pugui consultar directament els seus dubtes i que l'ajudi a prendre decisions de manera més àgil i informada.

A més d'aquesta línia principal, el treball obre altres direccions d'evolució. La primera és la d'incorporar un cicle de re-optimització periòdica: en un entorn real, la demanda, els terminis i els costos canvien amb el temps, de manera que els paràmetres haurien de recalcular-se de forma recurrent i no com un exercici puntual; l'arquitectura reproduïble de la *pipeline* ja està preparada per a aquest ús, però caldria definir-ne la cadència i els mecanismes d'alerta quan una recomanació canvia de manera significativa. La segona és la d'enriquir la simulació amb restriccions addicionals que avui queden fora del model, com ara les limitacions de capacitat d'emmagatzematge, els descomptes per volum del proveïdor o les dependències entre materials que es demanen conjuntament. Totes dues direccions comparteixen el mateix principi que ha guiat aquest treball: acostar progressivament un model matemàticament fonamentat a les condicions reals d'operació, sense renunciar a la traçabilitat ni a la reproductibilitat que el fan fiable.

AGRAÏMENTS

Vull agrair a Accenture l'oportunitat de desenvolupar aquest projecte durant la meva estada de pràctiques, així com la possibilitat de treballar en un entorn professional real que m'ha permès créixer i aplicar els coneixements adquirits durant el grau.

També vull donar les gràcies al meu tutor del TFG, no només per l'acompanyament, les orientacions i el suport durant l'elaboració d'aquest treball, sinó també per tot el que m'ha aportat al llarg del grau. La seva manera d'ensenyar, guiar i implicar-se ha estat una part molt important del meu aprenentatge.

REFERÈNCIES

[1] SAP SE. (2024). *SAP ERP Documentation – Materials Management and Inventory Management*. `https://help.sap.com`

[2] G. Hadley i T. M. Whitin. (1963). *Analysis of Inventory Systems*. Prentice Hall.

[3] F. W. Harris. (1913). "How Many Parts to Make at Once". *Factory, The Magazine of Management*, 10(2), 135–136.

[4] E. A. Silver, D. F. Pyke i D. J. Thomas. (2017). *Inventory and Production Management in Supply Chains*. CRC Press.

[5] Palantir Technologies. (2024). *Palantir Foundry Documentation*. `https://www.palantir.com/docs/foundry/`

[6] Apache Software Foundation. (2024). *PySpark Documentation*. `https://spark.apache.org/docs/latest/api/python/`

[7] Palantir Technologies. (2024). *Slate Documentation (Palantir Foundry)*. `https://www.palantir.com/docs/foundry/slate`

[8] A. M. Law. (2015). *Simulation Modeling and Analysis*. McGraw-Hill Education.

[9] J. Banks, J. S. Carson, B. L. Nelson i D. M. Nicol. (2010). *Discrete-Event System Simulation*. Pearson.

APÈNDIX

A1. TAULES SAP UTILITZADES COM A REFERÈNCIA

La *pipeline* pren com a referència l'estructura de les taules SAP següents:

TAULA 2: TAULES SAP DE REFERÈNCIA

Taula	Contingut
MSEG / MKPF	Moviments de material
MARA / MARC	Mestre de material
MAKT	Textos de material
EINA / EINE	Info. de compres (proveïdor)
DISPO	Controlador MRP
Storage loc.	Inventari per ubicació

A2. CAPTURES DEL RESUM EXECUTIU

Les figures següents mostren el resum d'optimització en detall (Fig. A1), el valor d'inventari (Fig. A2) i el desglossament de costos d'inventari: total, manteniment, comanda i *backlog* (Fig. A3–A6). La vista d'abast i segmentació i el resum executiu principal es presenten al cos del document (Fig. 2 i Fig. 3).



Fig. A1: Resum d'optimització (detall).

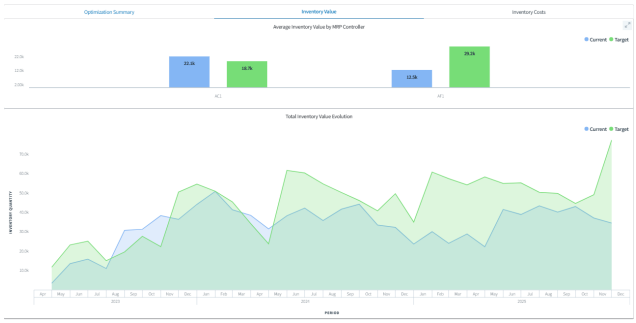

Fig. A2: Valor de l'inventari.



Fig. A3: Costos d'inventari: total.



Fig. A4: Cost de manteniment.

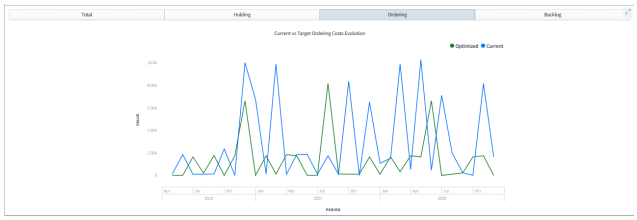

Fig. A5: Cost de comanda.

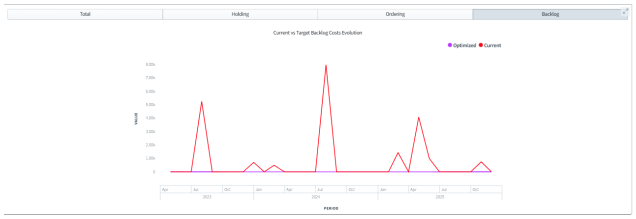

Fig. A6: Cost de *backlog*.

A3. DETALL DE MATERIAL I BACKTESTING

Les figures següents mostren la vista de detall d'un material (Fig. A7), l'evolució del valor d'inventari (Fig. A8) i les alertes de monitoratge, sortida i entrada (Fig. A9–A11); la vista de material amb demanda enfront de *forecast* i inventari enfront de demanda (Fig. A12–A14); i el *backtesting* de la projecció d'inventari (Fig. A15–A16). La projecció d'inventari *baseline* enfront d'optimitzat es presenta al cos del document (Fig. 4).

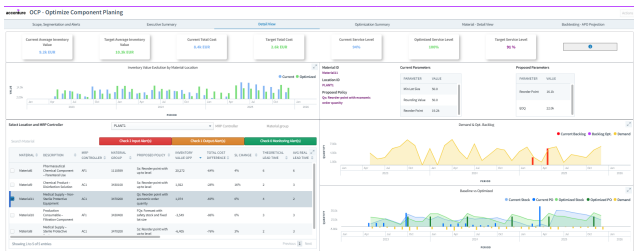

Fig. A7: Vista de detall.

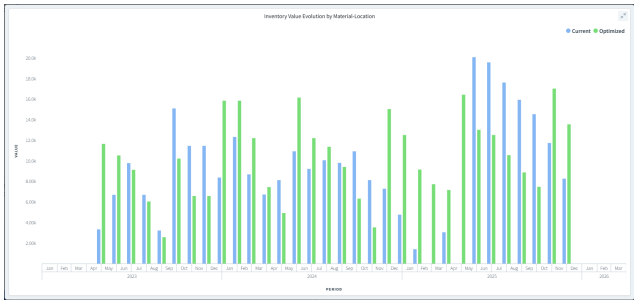

Fig. A8: Evolució del valor d'inventari.

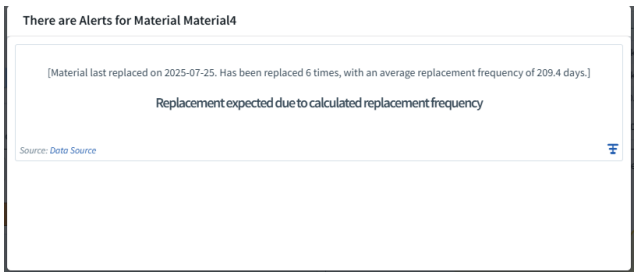

Fig. A9: Alertes de monitoratge.



Fig. A10: Alertes de sortida.

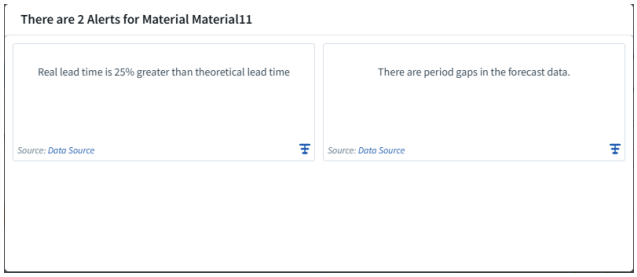

Fig. A11: Alertes d'entrada.

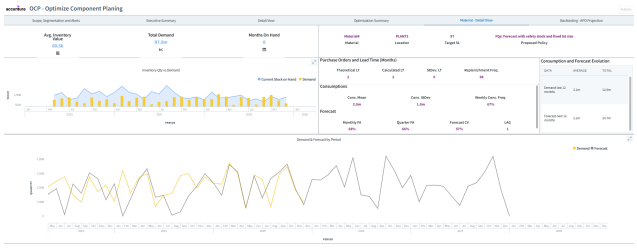

Fig. A12: Vista de material.

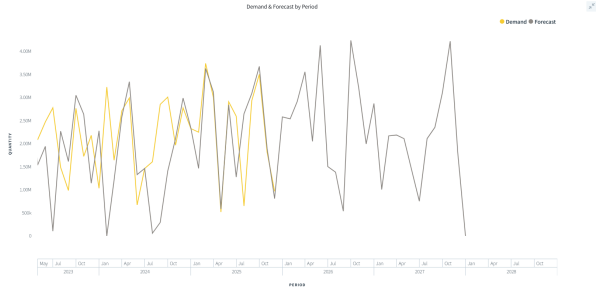

Fig. A13: Demanda vs *forecast*.



Fig. A14: Inventari vs demanda.

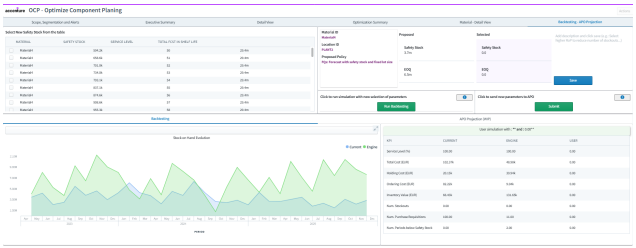

Fig. A15: *Backtesting*: projecció APO.

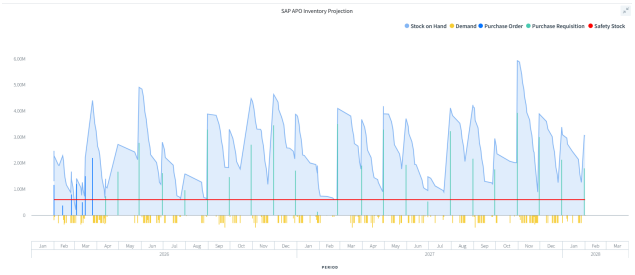

Fig. A16: *Backtesting*: projecció d'inventari sAPO/APO.