Faster, Cheaper, Better: Implementing Fusion Middleware at Capital & Coast District Health Board

Calum Laurie
Capital & Coast District Health Board

Greg Fookes
Hewlett Packard New Zealand

Introduction

Capital & Coast District Health Board (C&CDHB) needed to integrate the Oracle eBusiness Suite with their legacy applications, and they needed to do it fast. Hewlett-Packard (HP) recommended Fusion Middleware. CFO Calum Laurie explains, "We demanded an efficiency that the architecturally robust BPEL solution could deliver to help us achieve our goals."

There is plenty of slick marketing and hype around words and concepts such as Fusion, Business Process Execution Language (BPEL), and Services Oriented Architecture (SOA), but how easy is it to put it all into practice? How does it work in the real world with the eBusiness Suite? In addition, although persuading an IT professional of the merits of a SOA approach can be straight forward, convincing a CFO or CEO to spend money on middleware can be a hard sell. The inevitable questions needs to be addressed: will it save money? And, if so, how?

This paper details how BPEL was selected and used to successfully integrate with multiple legacy applications and complex flat file based data sources, allowing C&CDHB to achieve the benefits of a SOA approach. There are few aspects of this case study that address particularly complex or unusual business requirements, or highlight radically innovative applications of the technology. However, as C&CDHB's needs were simple, they are therefore shared by many organisations across all industries, so the applicability of the content is universal – how to do something better, in less time, AND save money: 'Faster, Cheaper, Better'.

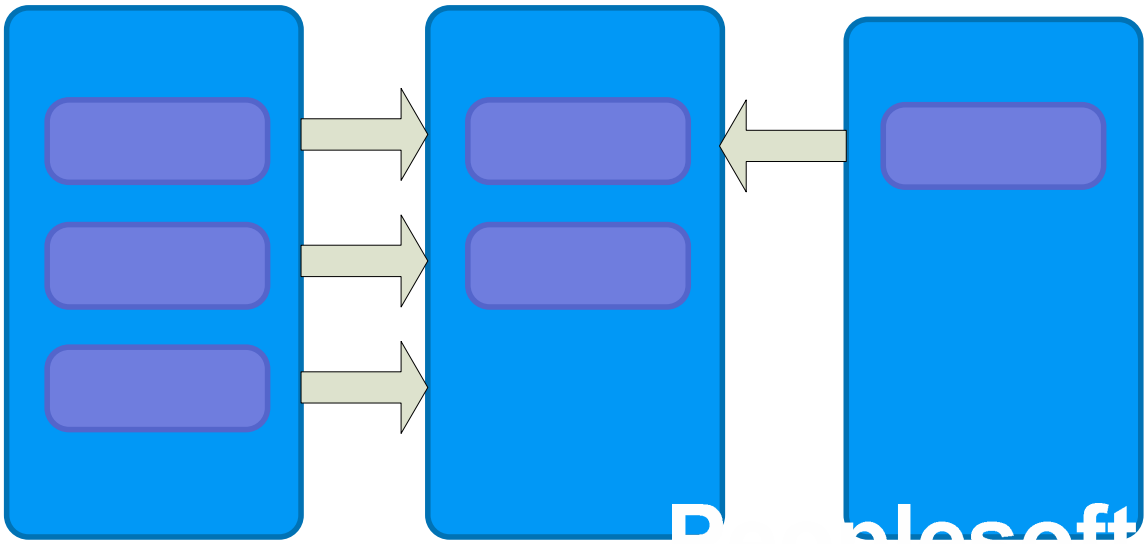
Capital & Coast District Health Board

Capital & Coast District Health Board <http://www.ccdhb.org.nz/> is a publicly funded, New Zealand based hospital and health care service. C&CDHB provides specialized hospital and community-based health services to the quarter-million residents of Wellington City and its suburbs, and to the greater central region of New Zealand. With 4,500 full-time staff and an annual turnover of US$550+ million, CCDHB is the largest employer in the Wellington region. The DHB also provides specialised care to the 900,000 residents of the lower North Island and upper South Island. These tertiary services include cardiology and cardiothoracic surgery, neurosurgery, vascular surgery, renal medicine and transplants, genetics, oncology, pediatric surgery, neonatal intensive care, obstetrics, endocrinology, orthopedics and urology, and specialized mental health and forensic services.

Project Background

By 2006, C&CDHB's legacy financial and materials management applications environment was proving to be unsuitable for the size of organisation that they had become. Based on a combination of a number of products supplied by different vendors, the complexity of the integration environment was difficult and expensive to maintain and was unable to be scaled as the needs of the business grew over time. C&CDHB were also lacking key features that would allow them to automate and streamline currently predominantly manual business processes. C&CDHB set out to find a fully integrated suite to replace their Peoplesoft, Chairman, and Hardcat systems (see Fig 1):

Figure 1: Scope of Systems Replacement

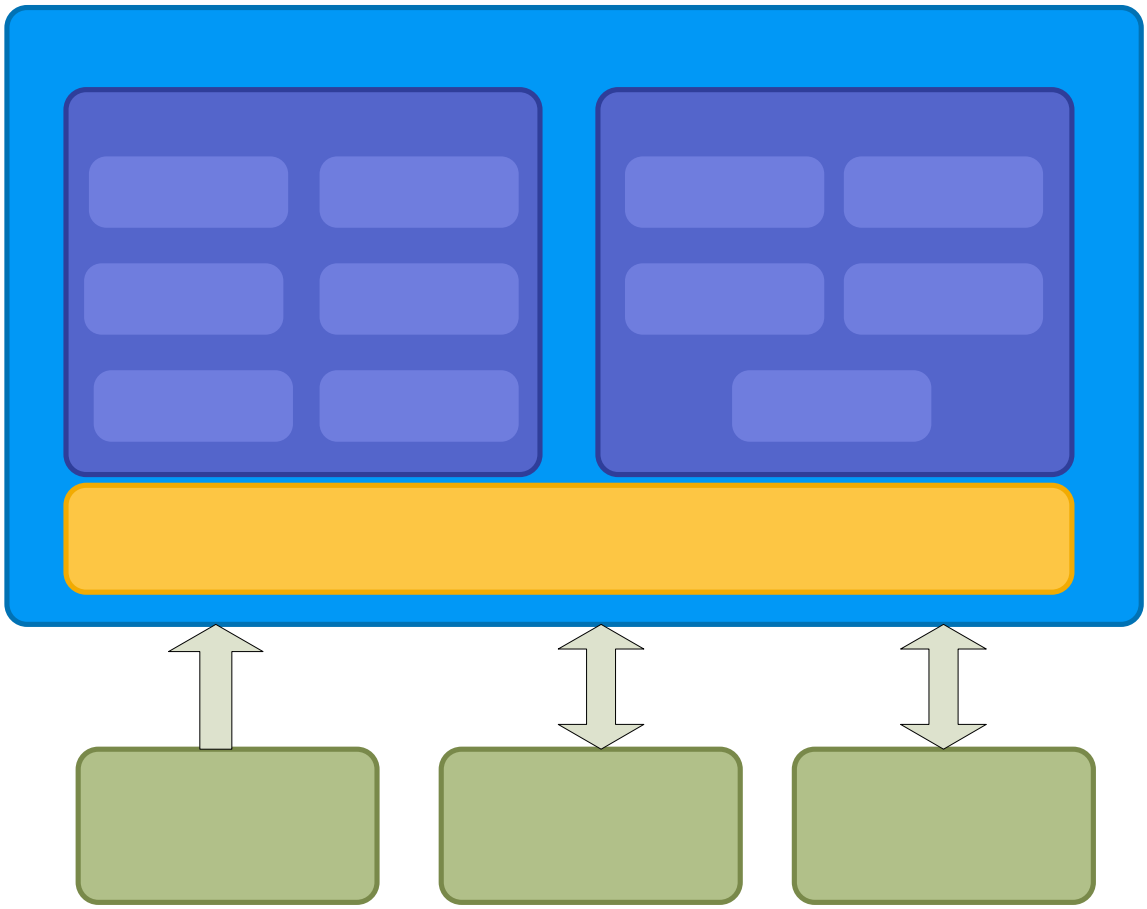


The new system would also need to provide them additional functionality in the areas of contract management, self service procurement, and business intelligence. After a thorough RFI and RFP process, C&CDHB selected the Oracle eBusiness Suite, with all aspects of the implementation including hardware and consulting to be provided by Hewlett Packard (HP). Oracle was fast becoming the standard for New Zealand's other District Health Boards, and HP were able to offer an accelerated implementation through the use of the intellectual property built up from implementing the eBusiness Suite for five other DHB's. With a planned duration of just six months, there was little time to spare when taking into account the usual requirements associated with a large systems replacement project. Further challenges for the Financial Management Information System (FMIS) project were added with the requirement to design and build new interfaces with three other systems.

Integration Requirements

The scope of the FMIS project included the development of five integration points with three key legacy systems that were to remain in use after go-live; a pharmacy system, a payroll system and a banking application (see Fig 2):

Figure 2: Future State Applications & BPEL Integration Scope

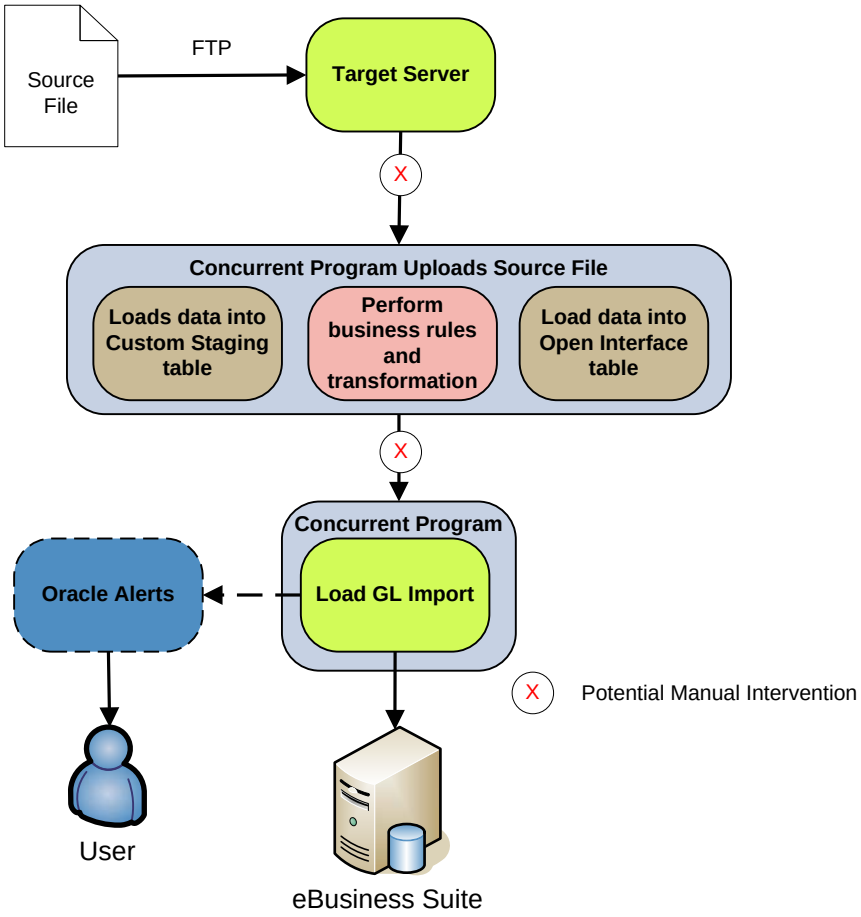


C&CDHB's key requirement for the integration was that the source systems had to remain unchanged. All data sources required complex mappings to proprietary data formats. Each interface required varying degrees of automation depending on the business need. C&CDHB also required that the interfaces be extensible, reliable, and easily maintainable.

Just prior to the commencement of the project, C&CDHB requested that the project timeframe be shortened to five months duration, to allow the system to be bedded in prior to the Christmas period. As part of the effort to reduce the overall project by one month, HP looked again at the approach for integration. In the absence of any licensed middleware technology, HP had previously performed the estimation for the development of the interfaces in accordance with the technology used by the other DHB's. Developed with only the source and target system in mind, this traditional approach resulted in a 'tightly coupled' point to point interface, utilising a combination of technologies to load and manipulate data into the formats required.

The diagram below shows a traditional GL interface where the source system transfers a file to the target system and multiple concurrent programs are executed to import the data into the eBusiness Suite (see Fig 3):

Figure 3: Traditional GL Interface



Typically the interface will be broken into three parts: 1) file transfer, 2) file mappings and business rules, and 3) execution of the open interface.

Some organisations may also develop some notification capabilities using Oracle Alerts to notify users of any exceptions that occurred during the import process. Organisations may also invest money to automate the whole process, but in most circumstances the business process will be to run the concurrent programs manually.

Traditionally this interface would have been developed using a combination of a number of tools such as custom FTP, SQL*Loader, PL/SQL, shell scripting, and Oracle Reports. These interfaces take significant time to get the various components to work together, and when things go wrong, diagnosis and correction usually requires a technical resource. The key disadvantage is if one of the systems were to change through replacement or upgrade, the interface could require a complete rewrite.

With time being critical, the use of Oracle's Fusion Middleware and particularly the BPEL component represented an opportunity to improve on the usability and architectural robustness of the integrated solution, at the same time as achieving some of the time savings necessary to meet C&CDHB's requirement for an earlier go-live. Ultimately, HP recommended the use BPEL as the business integration software because of its Service-Oriented Architecture (SOA) approach, which offers flexibility and ease of use to ensure compliance. Standards-based BPEL also demonstrated two other clear advantages: it would minimize conflicts between applications, as well as provide robust workflow capabilities to help automate complex business rules, which, with the traditional approach, would have been cumbersome to develop.

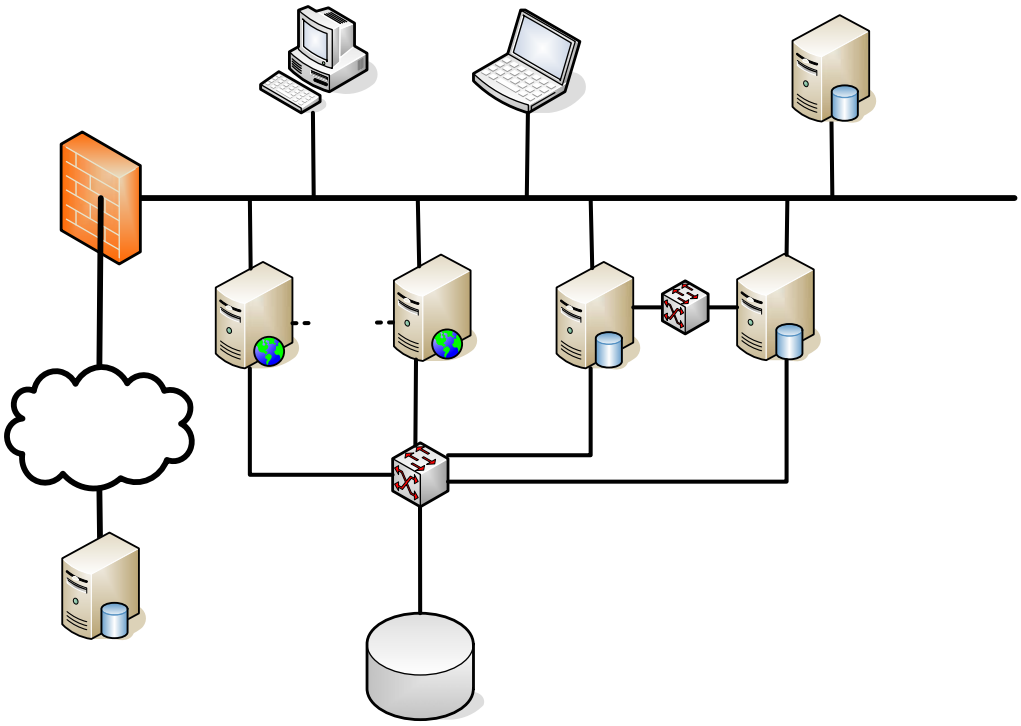
But how to pay for the software? An additional advantage of utilizing BPEL to integrate the eBusiness Suite with other applications was that the projected cost savings from reduced development effort could be applied to the license purchase. It was therefore straightforward for C&CDHB to justify the purchase on the basis that it would save money almost immediately, as well as result in a superior solution. As C&CDHB's future IT architectural strategy was already based on the SOA approach, it was also clear that there was significant future

value with the purchase. Oracle NZ responded quickly with pricing the licenses and the deal was concluded in time for the design work to begin on schedule.

Architecture Overview

One of the first aspects of the design that needed to be taken care of was that to ensure high reliability and performance, the BPEL Process Manager (BPEL PM) needed to be load-balanced over two HP Blade servers. The following configuration was confirmed (see Fig 4):

Figure 4: Physical Architecture

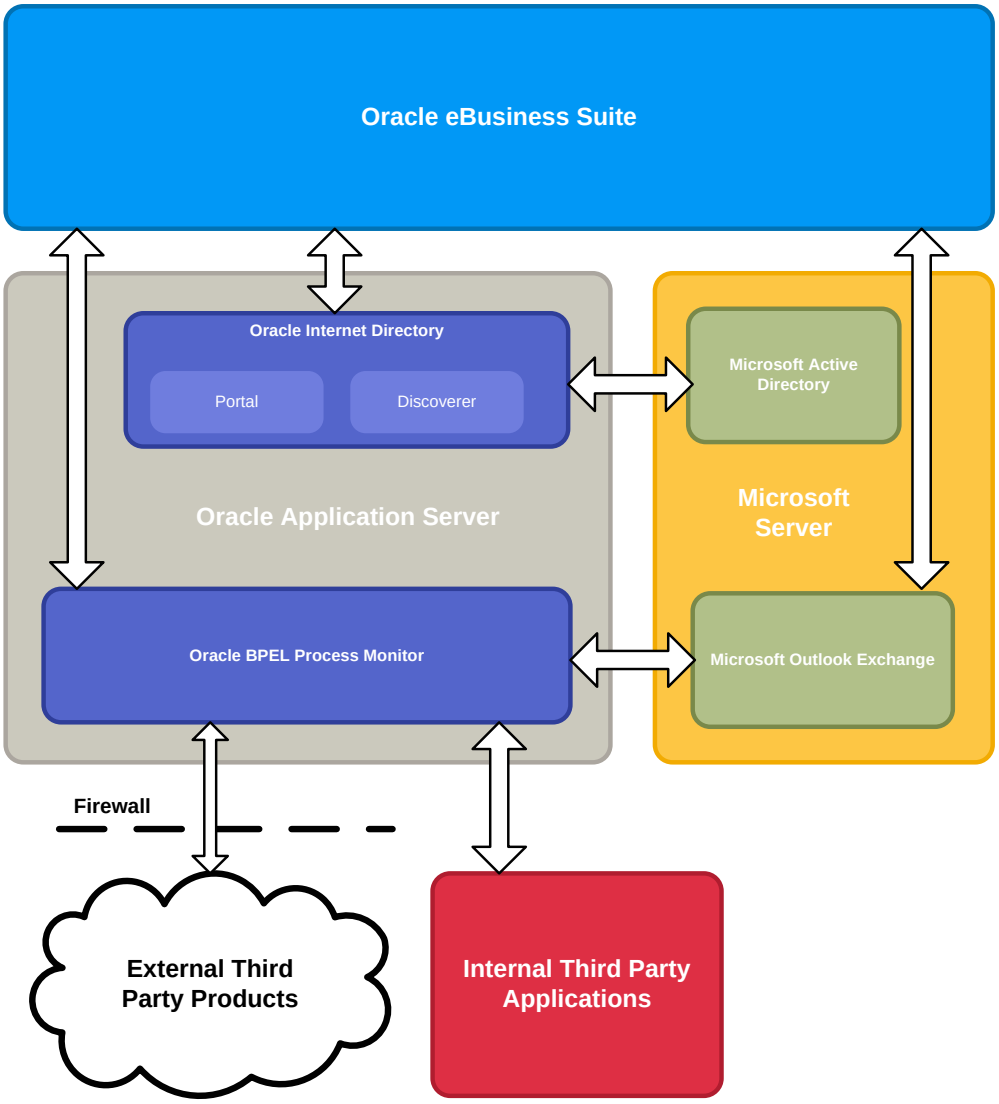


Workflow Automation

BPEL provides an advanced workflow engine. Many of the interfaces HP developed for C&CDHB required a workflow component to manage authorizations for particular files. Once data is received by the data source, a workflow authorization process is triggered and an email is sent to the designated employee. The BPEL process then halts all processing until the response message is received. Once the message is received, BPEL continues processing the authorization.

Email was the chosen delivery tool as it did not require any Oracle access by the user and notification was instantaneous. Using a worklist via a portal or standard eBusiness Suite workflow notifications risked causing a bottleneck in data processing as users typically used email more consistently than the eBusiness Suite. Due to the need to use notifications outside of the eBusiness Suite, the following integration with C&CDHB's Microsoft based email technology was implemented (see Fig 5):

Figure 5: High Level Design

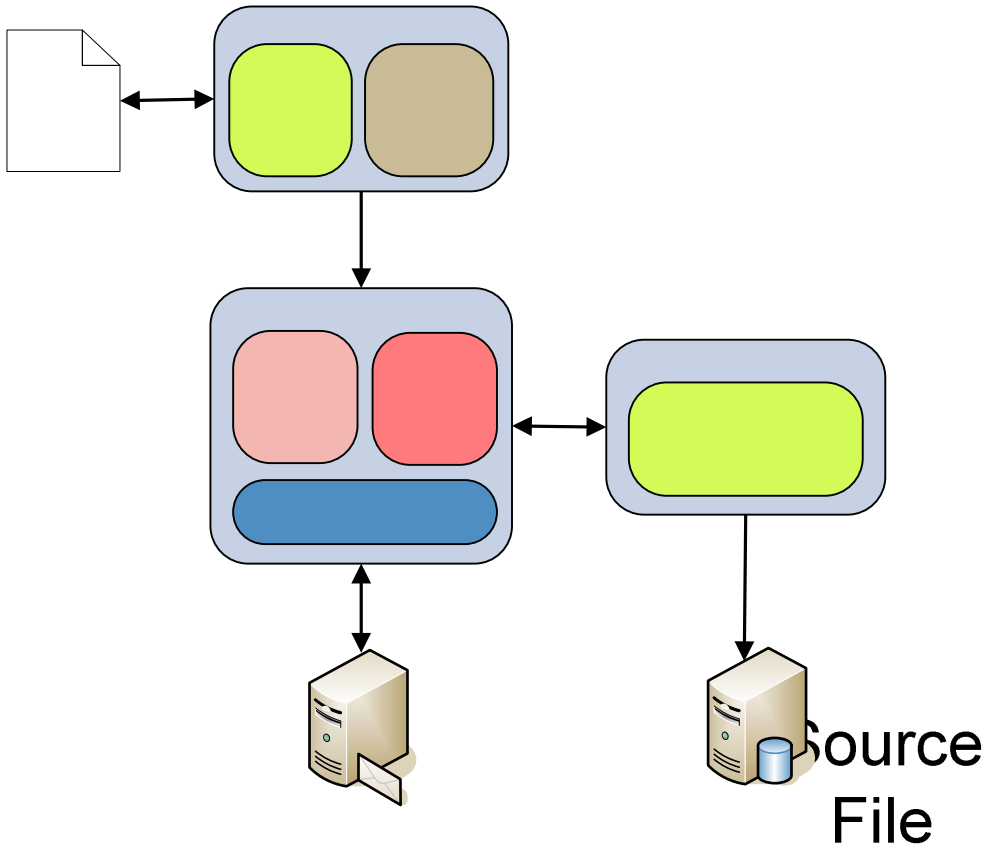


BPEL Process Design

The overall principle underpinning the design was that to be able to accommodate the ever-changing IT environment, SOA principles needed to be foremost to minimize implementation costs if source systems or data sources change in the future.

The design of the individual processes typically comprised of three BPEL services (see Fig 6):

Figure 6: BPEL Processes



In this design, the first BPEL service manages the inbound message and maps it to a generic mapping table. This service then calls the Main BPEL service. This service is where the business rules, workflow, and error handling is applied. The main purpose of this service is to translate source system codes into target system codes. This service also calls the Outbound Service which sends the message to the target system, in this case the Oracle eBusiness Suite. This architecture provided the flexibility so if the target or the source system changes it is just a matter of changing the Inbound or Outbound service without changing the core business logic defined in the Main BPEL Service.

This approach also enables the reuse of code so that multiple data sources can be used with one interface, such as a GL interface, as you can have multiple inbound services calling the one main BPEL Service.

Data Integration

Most of the data sources were complex flat files, which needed to be mapped into Oracle staging tables and communicated back to the Oracle eBusiness Suite via BPEL's database adapters.

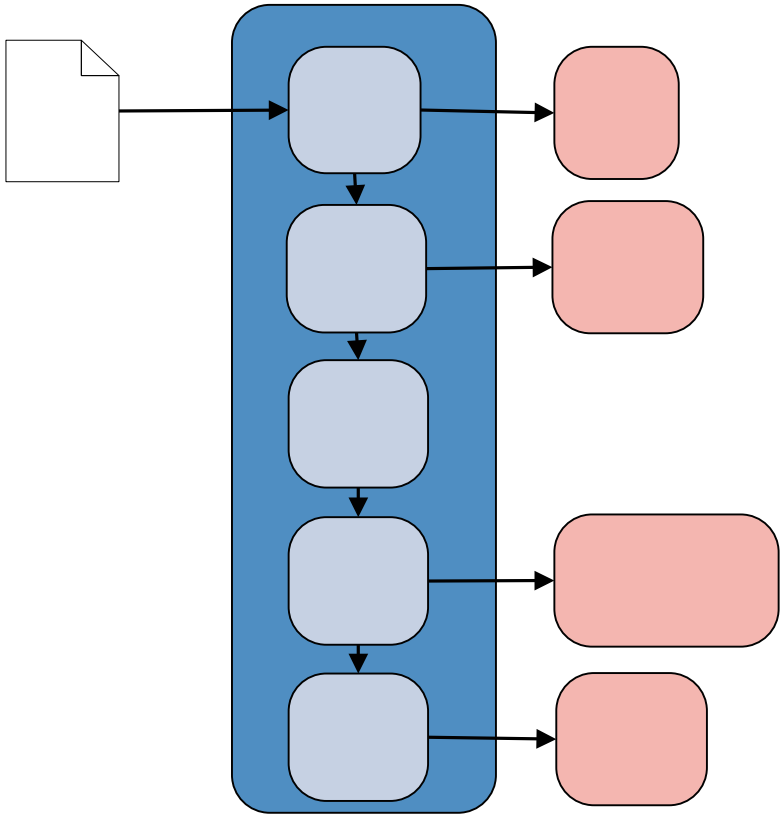
With the use of BPEL, the adapters are used to interpret messages from source systems. The messages are received and validated for data consistency. The data is then transformed into the common view to perform the business logic. With this approach at each step, data was loaded into staging areas so processes could be rolled back in case of failure.

Types of Integration Implemented

The implementation at C&CDHB needed three types of BPEL processes: Firstly, inbound transactions to use file adapters and database adapters; secondly, outbound transactions to be initiated via web services and distributed to the target system either by flat file or web service; and finally, inbound transactions requiring workflow. These transactions needed to interface with Microsoft Exchange server. BPEL needed to interpret emails received and perform the appropriate actions.

Integration with the WinDose system represented the most complex of the third party systems. WinDose is a pharmacy application used by C&CDHB, and includes functionality for dispensing, ward imprest, compounding, repackaging, supply management, stock control and reporting, and it required three points of integration: 1) a suppliers synchronization process with Oracle AP; 2) an invoice transfer from WinDose to Oracle AP; and 3) a transfer of summarized inventory cost data from WinDose to the Oracle General Ledger. The invoice transfer is a good example of an interface that needed to implement a BPEL process to accommodate inbound transactions using file adapters and database adapters. The process flow is provided below (see Fig 7):

Figure 7: WinDose Invoice Process

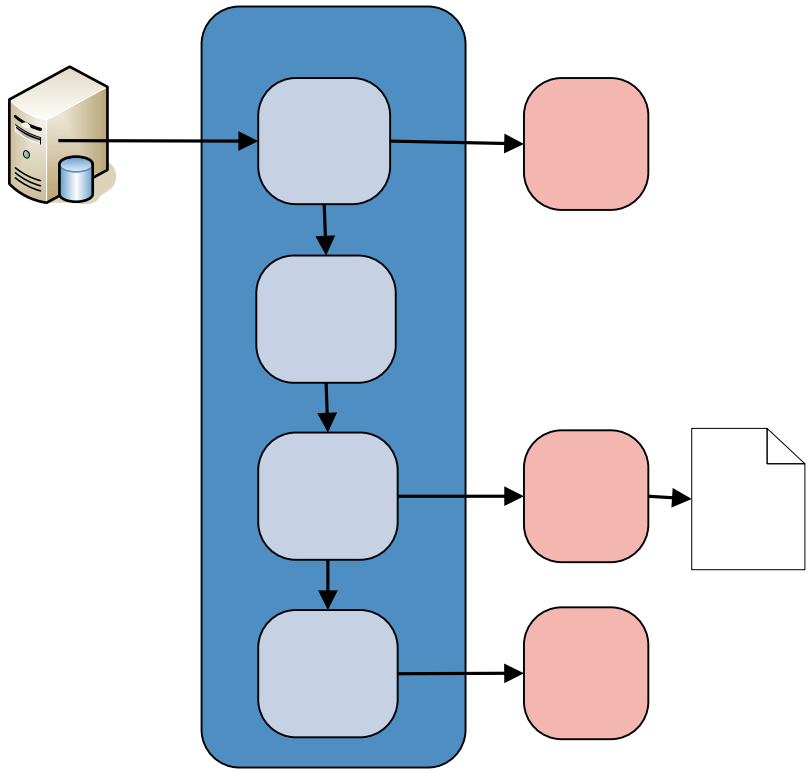


The file adapter picks up the source file from a network directory, by polling at frequent intervals (every 30 seconds). Once a file is received the BPEL process immediately executes, determines which user needs to be included in the process via a custom profile option stored in the database, transforms the file, and then inserts the data into staging tables. The validation logic from an earlier interface was able to be reused in a database procedure that then inserts data that meets the business rules into the standard open interface tables. Finally, a user is emailed the success or failure of the process with key financial information. A C&CDHB requirement was to keep the final step of loading the data into AP manual, so auto submission of the invoice import process was not implemented.

An additional integration point was required to synchronise the supplier data in Oracle with the suppliers in WinDose – if these did not match then invoice files would not validate and therefore fail, requiring manual intervention. In this interface, Oracle is the initiation point – by calling the BPEL web service via a concurrent program, the process extracts and transforms the data into the format required, and then places the file in the directory that had been in use with the previous Peoplesoft system (see Fig 8):

Source File

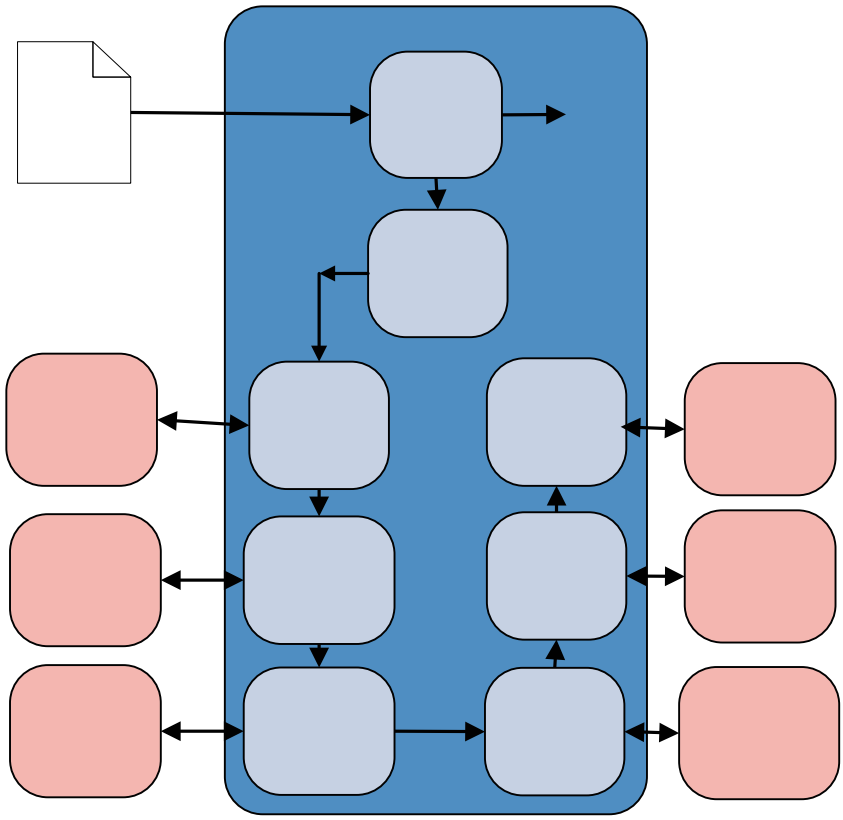
Figure 8: WinDose Suppliers Process



By implementing in this way, the change in system was completely transparent to WinDose. If in the future WinDose was replaced, only the mapping to the output file or system would need to change. One difficulty was found with this interface – the intention in the design was for the initiating concurrent program to be synchronous in nature i.e once the program initiated the BPEL process via the call to the web service, the program would wait for success or failure so it could complete with the necessary status. This proved difficult to achieve (even with Oracle's help) so the fall back plan was implemented – an asynchronous process was used. In this case the calling program always completed successfully regardless of the execution of the BPEL process. Errors need to be detected solely via the BPEL console.

The final type of interface implemented was for inbound transactions that needed to use BPEL's workflow capabilities together with Microsoft Exchange integration to request and receive decisions from end users via email, and then process data appropriately. The GL interface was implemented to process files from two systems: WinDose, and LEADER Payroll. Leader is an application for the health sector in New Zealand encompassing aspects of rostering, contract and award interpretation, payroll, and human resource management. The interface was implemented using the following flow (see Fig 9):

Figure 9: WinDose & Leader GL Process



The key differentiator for this interface is that it allows for two separate file formats to accommodate both systems that need to use it, and also it uses email responses to determine which subsequent actions to take. The design of the interface allows for straightforward extensibility – if C&CDHB wish to add a new source of GL data, they only need to modify the inbound BPEL service, leaving the other program components intact. The email processing alerts users that a file has been received and is in process. The user is given key financial information about the inbound file and asked if accrual is required. Based on the response, the BPEL process inserts data into the core GL interface tables, and then submits the GL import process automatically. At the conclusion of the import, the user is notified of the completion of the interface. From start to finish, the only user involvement necessary is a simple email response.

Faster, Cheaper, Better?

So was it all worth it? Did C&CDHB achieve the benefits they set out to achieve? When C&CDHB set out to build their interfaces they were looking to satisfy three clear goals. Firstly, it had to be implemented quickly. The project was seeking to achieve a great deal in a very restrictive timeframe, and time savings had to be found in all project streams. Secondly, the solution had to have a compelling cost vs benefit ratio in order for the purchase decision to be vindicated. Finally, whatever approach was selected, it had to meet the requirements for reliability and automation that was lacking in their legacy system.

The use of Oracle BPEL at C&CDHB saved the time required to complete all integration work before the transition to go-live. The graphically based development tool was well loved by the HP development team, especially when the alternative was trying to find a missing full stop or bracket in a shell script. Data transformations or mappings can be implemented in minutes rather than the hours required in developing a script. Although limitations in the ability of BPEL to perform sophisticated queries does not remove the need to build views or procedures in the database, the HP team determined that the overall time required to implement a BPEL interface should be around half of that required for a traditional interface where multiple technologies are involved.

In addition to saving time, the use of BPEL proved to be a financially sound decision. It's probably most appropriate that the final word on the financial decision comes from the C&CDHB staff member that is most vested in ensuring that money is well spent. CFO Calum Laurie says "HP helped us decide that this solution would save us development time and cost. It did exactly what we needed: It speeded up development and has reduced development costs for C&CDHB. In fact, it has already paid for itself over these five interfaces."

Finally, there is no question that the SOA enabled approach offered by BPEL resulted in an overall superior solution than what would have been achieved had the traditional approach been followed. The solution is more robust and reliable. The failover offered by the BPEL process manager being load balanced across two servers is far superior to traditional concurrent manager bound processes where the manager is most often located on a single database server. The BPEL console puts the problem resolution process back in the hands of a non-technical systems administrator. Problem processes can be easily restarted from the point of failure, instead of having to manually clear out database tables, and then try to figure out what should be run next. A better balance has been struck between automation and control with the use of email to notify and drive workflow based business rules through the processes. Furthermore, the SOA approach has ensured that future developments in C&CDHB's application roadmap will not necessitate wholesale rewrites of integration code. C&CDHB have just begun with the use of BPEL and the other components of the Fusion technology stack. With the wide range of adapters available, C&CDHB have a great deal of flexibility and capability when considering future integration projects.