

ASD SaaS Business Case for Philips Healthcare

Author: Robert C. Howe
Version: 1.0



The simple way to build
complex software
systems

EXECUTIVE SUMMARY

Philips Healthcare Cardiovascular division has been using Verum's ASD:Suite for production software development since April 2008. In September 2008, Philips started using Verum's first early customer release of the ASD:Suite. This release collects data about how customers use the ASD:Suite.

Based on data collected during October and November 2008, Verum has been able to analyze Philips' use of the ASD:Suite and has been able to compare and contrast these figures with industry standard data. In the period April to November 2008, Philips produced 27381 executable lines of code using the ASD:Suite. Based on best case assumptions, the conventional cost of developing this amount of software would have been €848,811.-.

A conservative extrapolation of actual data indicates that, according to Verum's Software as a Service (SaaS) business model, the total cost of developing this software with the ASD:Suite would have been €541,161.-, a cost saving of 36%.

Sensitivity analysis shows that in all likely scenarios, the application of the ASD:Suite would have resulted in double digit cost savings over conventional development methods. Further, software developed with the ASD:Suite is verifiably defect free, whilst software developed conventionally is not.

This analysis clearly demonstrates the business case for Verum's ASD:Suite at Philips Healthcare.

1 INTRODUCTION

1.1 PURPOSE

This paper presents the business case for using Verum's ASD:Suite at Philips Healthcare Cardiovascular division at Best in the Netherlands. The case is based on an analysis of the actual use of the ASD:Suite made by Philips during the period September 2008 to December 2008, using data gathered from the ASD:Suite metrics database.

1.2 BACKGROUND INFORMATION

Verum's ASD:Suite is a toolset for rapidly designing, formally verifying and automatically generating software for complex software systems. ASD:Suite delivers demonstrably high reliability software for a lower cost per line of code than conventional software development methods. ASD:Suite is part desktop application, part hosted service and is offered to customers on a Software-as-a-Service (SaaS) basis, resulting in further cost savings. ASD:Suite collects metrics about the way that customers use the various ASD tools. These metrics can be used to provide an objective measure of the use that customers make of the ASD:Suite and to indicate the value that it delivers.

For the past three years, Philips Healthcare has partnered with Verum in the development of the ASD:Suite. Philips started using the ASD:Suite for production software development in April 2008. Since September 2008, Philips has been using an Early Customer release of the SaaS ASD:Suite product. Based on metrics collected from September to December 2008 it has been possible to provide Philips with an indication of the value that they have had from the ASD:Suite. This paper details the resultant business case.

2 ASD:SUITE SAAS BUSINESS MODEL

2.1 BACKGROUND

Verum's ASD:Suite product consists of a number of components:

Desktop product	Hosted product	Function
ASD:ModelBuilder		Provides graphical tools for building software interface and design models and for verifying that interfaces, designs and related specifications are complete and correct.
	ASD:ModelChecker	Enables models to be mathematically verified and identifies any potential defects in interfaces or designs, so that they can be removed before the software is generated
	ASD:CodeGenerator	Automatically generates source code. Typically 90% of a design's code can be generated automatically. Currently supported target source codes are C++, C#, C and Java
	ASD:Portal	User administration, licensing, billing, subscription management

The ASD:ModelBuilder is the user's gateway to the other ASD toolset components. It fully integrates access to the ASD:ModelChecker and the ASD:CodeGenerator. The latter components are hosted on dedicated server hardware at a Data Centre. Since the ASD:ModelChecker is a CPU intensive application, Data Centre hosting ensures that customers are provided with a high performance, high availability ASD computing resource that requires minimum system management by the customer. Further, customers always have access to the latest version of ASD:Suite components and, perhaps more importantly, to legacy versions.

Via the Verum website, the ASD:Portal provides customers with access to user administration, licensing, billing and subscription management facilities.

2.2 ASD BUSINESS MODEL

Verum's business model for the ASD:Suite is similar to a mobile phone business model: customers pay a license fee for the ASD:ModelBuilder and a volume related charge for use of the hosted ASD:ModelChecker and ASD:CodeGenerator. Use of the ASD:Portal for the aforementioned features is free (for customers).

The ASD:ModelChecker license fee is an annual payment that covers use of the model checker, upgrades to all ASD:Suite components, and maintenance and support.

The volume related charge is calculated based on the size & complexity of the models that the customer builds and the number of times the model(s) are model checked or used for code generation. Model size & complexity uses an objective unit definition referred to as an ASD Function Point (FP).

2.3 ASD FUNCTION POINT DEFINITION

ASD models possess certain attributes that scale with the size & complexity of the models. Verum has defined an objective formula for an ASD Function Point, based on counting model attributes. Further, the definition has been scaled such that 1 ASD Function Point \approx 1 ELOC¹ C++. In other words: an ASD model of size 1000 ASD Function Points produces approximately 1000 ELOCs of C++ when run through the ASD:CodeGenerator.

The ASD Function Point is the (objective) metric upon which volume related billing is based. Verum publishes the definition of an ASD Function Point for (curious) customers.

2.4 ASD VOLUME RELATED BILLING IN PRACTICE

To clarify how ASD volume related billing works in practice, consider the following (simplified) situation:

1. A customer has designed an ASD Model that consists of a design and two interface definitions.

¹ ELOC = Executable Line of Code

2. The design model has a size of 1600 ASD FPs and the interface models are each 200 ASD FPs. The total component design is therefore 2000 ASD FPs.
3. Assume that it takes the customer 19 iterations through the model builder before the design & interface models are shown to be correct (zero defects).
4. Assume that the customer then generates C++ from the complete & correct interface and design models.

In this case the customer would have consumed:

2000 ASD FPs x 20 hosted service interactions = 40000 ASD units

This activity would have resulted in approximately 2000 ELOCS of (defect free) C++ as a result.

Of course, reality is a little different: most software designers work incrementally and therefore the size of an ASD model tends to grow over until it reaches a final size. Thus, in practice FP volume consumption would most likely be lower than stated in this simplified example.

2.5 UNSUBSCRIBED AND SUBSCRIBED VOLUME

Like a mobile phone operator, Verum offers two ways to buy ASD FP volume: customers can buy volume on an adhoc, pay-as-you-go basis or on a subscription basis or a combination of both. Subscribed volume costs 33% less than unsubscribed volume, but is committed for the period of the license contract. For more details on ASD pricing, please contact Verum's Sales Department.

2.6 ASD METRICS

In order to bill customers, the ASD:Suite collects the following metrics per customer:

1. The number of ASD:ModelBuilder/ASD Server transactions
2. The date and time of each transaction

Verum white paper study

ASD SaaS Business Case for Philips Healthcare

3. The ASD FP volume of each transaction
4. The type of the transaction (model check, code generate, etc)
5. If code generated: the target language (C++, C#)
6. The name of the associated interface or design model
7. The name of the user associated with the transaction

Based on these metrics it is possible to calculate not only the customer's bill, but also an indication of how much value they obtained from using ASD. The next chapter discusses the metrics results and value obtained by Philips Healthcare.

3 Philips Healthcare ASD Metrics and Value

3.1 Background

Philips Healthcare has partnered with Verum for the last two and a half years in the development of the ASD:Suite. Since April 2008 Philips has used early customer versions of the ASD:Suite for the development of production software. In September 2008, Verum released a SaaS based version of the ASD:Suite for early customers. Philips began using this version at the end of September.

In December 2008, Verum analyzed Philips' use of the ASD:Suite based on the two month's worth of ASD metrics data that had been collected at that time. The results of this analysis are presented below.

3.2 Philips Healthcare ASD Metrics

3.2.1 Development Teams

In the period concerned essentially two different software development teams made use of the ASD:Suite. The first team consisted of 1.5 software designers who were developing complex C++ components for a product. The second team consisted of 5 software designers developing C# components for a different product.

It is important to note that in this case the ASD metrics provided only a partial view of the activities of each development team: team 1 started using the ASD:Suite in April 2008 and team 2 in June 2008.

3.2.2 Results

In total the two development teams consumed 1,234,006 ASD FPs in 1364 ASD server interactions. The average model size was therefore 923 ASD FPs. Figure 1, page 14, shows ASD FP volume broken down by activity. Around 60% of total ASD FP volume was used for model checking and 40% for code generation.

Verum white paper study

ASD SaaS Business Case for Philips Healthcare

Further analysis produced the following table:

Philips results for 2 months					
	LOCs	Users	Man days	ModelChecker Defects	FP Volume
C++	7697	1.5	10	36	227623
C#	19684	5	69	387	1006383

At the end of November, Team 1 had produced 7697 ELOCs of C++. In the preceding two months they had used ASD for a total of 10 man days and found 36 defects through model checking. Note that 7697 ELOCs is the total amount of code produced since the start of the project in April. Team 2 had produced 19684 ELOCs of C# from May to the end of November using ASD. In the two months during which data was collected they found 387 defects in 69 man days of effort.

Note that the current ASD metrics implementation does not include a direct measure of effort. To obtain an effort figure a conservative assumption was made: any day on which a user executed a single transaction with the ASD server was counted as one man day of ASD effort.

Note also that ASD metrics do not yet include a direct measure of the number of defects found during a single model check. Therefore a conservative assumption was made: one model check = one defect. In general this assumption corresponds with practice.

Finally, it is also worth noting that these figures were collected during a period when Philips designers were learning to use ASD and to apply it to various types of problems. It is expected that as they become more accustomed to ASD their productivity will rise.

3.3 Comparison with Conventional Methods

3.3.1 Assumptions

In order to derive the business case for the ASD:Suite, one needs to compare and contrast the cost of using ASD with the cost of developing the same amount of software using conventional means. In order to do this a number of assumptions must be made:

Verum white paper study

ASD SaaS Business Case for Philips Healthcare

Assumptions			
Metric	Assumption	Typical Range	Notes
ELOC / man hour	3	1 - 3	Supported by studies
Cost / Man Year	€120,000,-		Typical for region
Utilization	80%	60-80%	
Costs / Man Hour	€75,-		
Defects / KELOC	3	2 - 8	Supported by studies
Average Cost of Defect	€2000,-	Upwards	Lowest likely cost

Studies that have looked into the subject of software productivity and defect rates (eg REIF01²) have shown that technical software development projects typically produce 1-3 ELOC/Man Hour. Defect rates at system testing range from 2-8 defects per thousand ELOC. Since Philips Healthcare has a mature software development process it is fair to assume best case figures of 3 ELOC/man hour and 3 Defects / KELOC.

Equally, best case assumptions have been made for utilization and average defect cost. The cost of a man year is typical for the industry in this region. **One important point to note is that these figures are assumptions about Philips Healthcare's performance and not actual figures.** For obvious reasons, Philips' actual performance figures remain confidential. However Philips has agreed that in their experience the above figures are typical.

3.3.2 Conventional Cost

Based on these assumptions it is possible to calculate the cost of conventionally developing the code volumes reported in section 3.2.2 in the same period of time. The following table shows these costs:

² RIEF01: Industry Software Cost, Quality and Productivity Benchmarks
Donald J. Reifer, Reifer Consultants Inc. April 2004.

Verum white paper study

ASD SaaS Business Case for Philips Healthcare

Conventional Cost Estimate (Best Case)								
	ELOCs	Users	Effort (Man Hours)	Duration (Months)	Defects	Development Cost	Cost of Defects	Total Cost
C++	7697	2.5	2566	8	23	€192,425	€46,182	€ 238,607
C#	19684	8.5	6561	6	59	€492,100	€118,104	€ 610,204

To produce 7697 ELOCs of C++ using conventional methods would cost €192,425.-. Conventional rework to remove defects would cost an additional €46,182.-. For C#, 19684 ELOCs would cost €492,100.- with an additional €118,104 in rework due to defects. In both cases the cost of a delivered line of code is approximately €31,-.

3.3.3 ASD:Suite Cost

Based on Verum's SaaS business model, the cost of producing the same amount of software with the ASD:Suite would be:

ASD Cost Estimate								
	ELOCs	Users	Effort (Man Hours)	Duration (Months)	Defects	ASD Cost	Effort Cost	Total Cost
C++	7697	1.5	320	8	0	€91,049	€24,000	€115,049
C#	19684	5.0	1656	6	0	€301,915	€124,200	€426,115

The table shows a split between the cost of using the ASD:Suite and the cost of the associated effort. These figures are based on the (conservative) assumption that Philips rate of work remained the same during both ASD project. However it is likely that initial rates of work were lower due to the time it took the designers to learn to apply ASD.

Note that there is no rework associated with the use of the ASD:Suite. Note that the C++ team delivered 24 ELOCS / man hour and the C# team 12 ELOCS / man hour.

3.4 Results

Using best case assumptions, the conventional cost of these two software development projects would have amounted to €848,811.-. Using the ASD:Suite the total cost would have been €541,161.-, a cost saving of 36%.

3.5 Sensitivity Analysis

The following table shows how the cost benefit of using ASD varies with a range of ELOCS per man hour and Defects per 1000 ELOCS. It can be seen that in almost all but the most exceptionally efficient and high quality environments, ASD offers the customer considerable cost savings.

ASD Value Sensitivity Analysis									
	Defects per kELOC								
	0	1	2	3	4	5	6	7	8
Delivered ELOCs per man hour	0.5	87%	87%	87%	87%	88%	88%	88%	88%
	1	74%	75%	76%	76%	77%	77%	78%	78%
	1.5	62%	63%	65%	66%	67%	68%	69%	70%
	2.0	50%	52%	55%	57%	58%	60%	62%	63%
	2.5	38%	42%	45%	48%	51%	53%	55%	57%
	3	27%	32%	36%	40%	44%	47%	49%	52%
	3.5	16%	22%	28%	33%	37%	41%	44%	47%
	4	5%	13%	20%	26%	31%	36%	40%	43%
	4.5	-6%	4%	13%	20%	26%	31%	36%	39%
	5	-16%	-4%	6%	14%	21%	27%	32%	36%

4 Conclusion

In the period April to November 2008, Philips produced 27381 executable lines of code using the ASD:Suite. Based on best case assumptions, the conventional cost of developing this amount of software would have been €848,811.-.

A conservative extrapolation of actual data indicates that, according to Verum's Software as a Service (SaaS) business model, the total cost of developing this software with the ASD:Suite would have been €541,161.-, a cost saving of 36%.

Sensitivity analysis shows that in all likely scenarios, the application of the ASD:Suite would have resulted in double digit cost savings over conventional development methods. Further, software developed with the ASD:Suite is verifiably defect free, whilst software developed conventionally is not.

This analysis clearly demonstrates the business case for the ASD:Suite at Philips Healthcare.

5 Appendix 1: Figures and Graphs

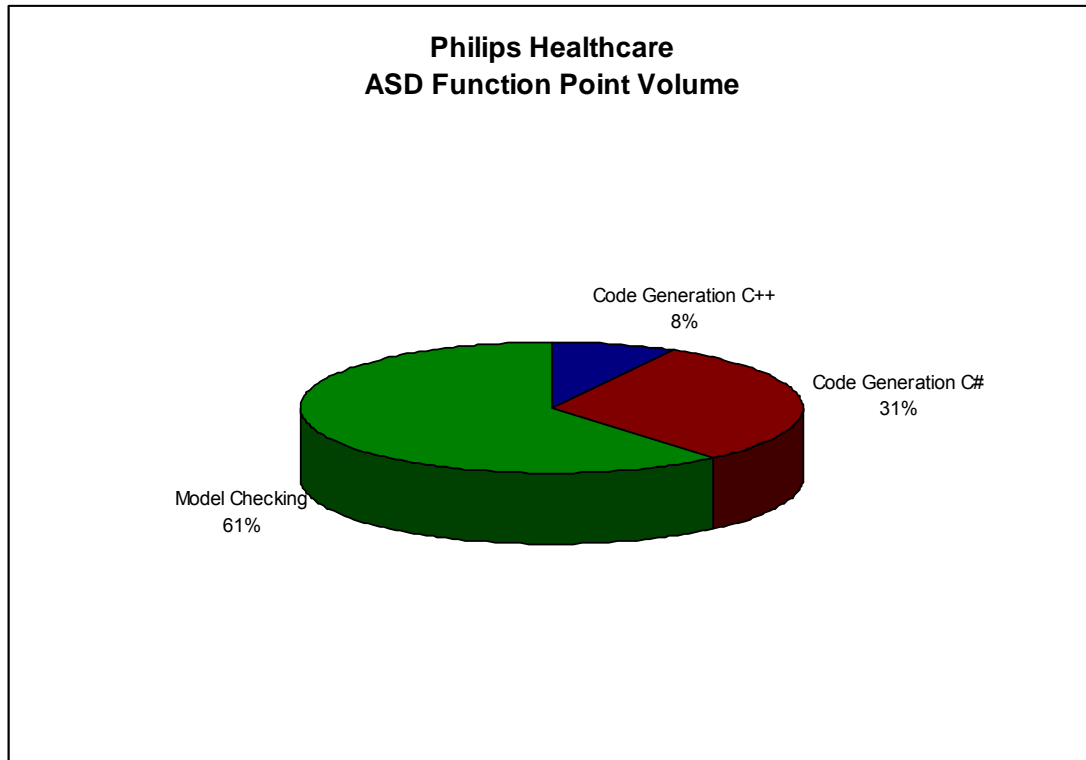


Figure 1 ASD Server Volume