BPEL Metrics in Large Scale Process Integration – A Time-Line

Daniel Lübke

innoQ Schweiz GmbH daniel.luebke@innoq.com

Abstract. I am now more than two years at a client's large-scale business process integration project, in which we integrate both administrative and commercial software systems for providing end-to-end and fully electronic business process support for the 80% standard use cases. The system has developed over time, new functional blocks have been added and I was curious to see what code metrics we have – especially because I have been asked many times how "complex" or "large" our project is. Because noone I have spoken to so far has an idea what a "typical" BPEL process size is this might be a first step towards a metrics comparison for "real" BPEL processes (in contrast to all the simple book examples.) Therefore, in this technical report I present the code metrics and mutation metrics of our project over the last months.

1 Introduction

This technical report describes the BPEL metric values for the project I am currently working at. In this project BPEL (used as an abbreviation for WS-BPEL [5] in this report) is used to orchestrate internal services and partner services to accomplish an end-to-end, fullly electronic process support between administrative and commercial parties. The developed system is acting as an integration hub, routing and integrating data and controlling the business process flow.

These are the interesting process properties that need to be considered when interpreting the metrics' values:

Process Scope: Cross-Organizational/B2A,
Process Type: Integration,
Integration Depth: System-to-System, planned with up to 1000 systems,
Automation: Full Automation for Main Processes and nearly full automation for Business Functions (see below), nearly fully manual for UI processes,
Process Depth: 80-90% case, not full coverage. This means that special rules are sometimes not supported in order to reduce complexity. In such cases the process has to done on paper as it is conducted up to today,
Fault Management: Out of Process, i.e. the processes do not include automatic fault management. Whenever a process instance has a problem it is suspended and the problems are solved out-of-process. The process instance is then either fixed and resumed or terminated.

Both Process Depth and Fault Management characteristics lead to process models that contain nearly none of the fault and compensation managing activities like `<catchAll>`, `<catch>` and `<compensationHandler>`.

The project uses ActiveVOS [4] as the Business Process Management System (BPMS) which uses BPEL as the technical process language. ActiveVOS supports BPEL4People [1] and has properitary support for XQuery. In addition the editor creates BPEL `<scope>` and `<sequence>` activities around message passing activities (`<invoke>`, `<receive>`, and `<reply>`) that are not visibile to the process modeler in order to make the process visual representation cleaner by hiding the often occurring `<assign>-<invoke>-<assign>` structure. However, this means that much more `<sequence>`s and `<scope>`s are used than if another BPEL modeler would be used.

For testing BPELUnit [8, 7] is used that can provide mock services during a unit test and in addition offers integrated deployment, and code coverage metrics [6].

There are several types of processes which have different characteristics:

Main Processes: Top-Level Business Processes only contain business-related logic which is orchestrated out of SOAP Web services,

Business Functions: Reusable Sub-Processes implemented in BPEL that are used in several main processes,

UI Processes: For partners that do not integrate their own system, the integration platform allows the user access with a Web portal. In this Web portal a process complimentary to the main processes to be executed is implemented to control the partner's interactions. The long-term goal is that these processes are implemented on the side of the partner in their partner systems and are not needed anymore on the integration platform at all. These processes also contain BPEL4People tasks for managing user interaction. Besides these "portal processes" two processes exist for simple human workflows that allow people to sign off activities that are then booked via Web services to back-end systems,

Technical Helpers: Small processes that serve a dedicated, technical only purpose. Typically these could have been implemented as a Java service as well but the implementation was easier in BPEL due to its inherent Web service support,

Authorization Handlers: Because we needed per-instance authorization for SOAP messages authorization handlers, which are like aspects in Java, push permissions to the ESB layer whenever a process instance is created or finished,

Test Mocks: Some processes simulate partner behaviour and can be used for testing by addressing the mock partner instead of a real partner. These mocks are not BPELUnit mocks which are only available during a unit test but are persistent and installed on the test stage,

Prototype Processes: For testing technical interaction with other services.

Out of these categories this paper will show metrics for the first three types of processes: Main Processes, Business Functions, and UI Processes. These re-

semble real business processes and are not (most often very simple) technical implementations.

The processes are anonymized and will be refered to as P1..7 (for Main Processes), BF1..3 (for Business Functions), and UI1..5 (for UI Processes.)

2 Metrics Gathered

For answering how "large" and "complex" our project is, a set of easy to calculate metrics was chosen. The project does not sponsor these activities so a prerequisite for every chosen metric was:

- An Open-Source implementation must be available
- It must be quick to calculate and not block any servers for a long time

Because of this, there are currently no Control-Flow Complexity (CFC) [2] metrics because there is no tool support available. They have also the drawback that activities out of the main control-flow, e.g. in compensation handlers or fault handlers are not included in the CFC metric, thereby leaving out one important aspect of BPEL process complexity.

Instead activity counts and mutation counts are provided. Activity counts were gathered with a small script that later evolved into a BPELUnit subproject and mutation counts were calculated with the muBPEL [9] tool. These metrics have been finally be chosen:

Basic Activity Count: The aggregated count of all basic activities (`<assign>`, `<compensate>`, `<compensateScope>`, `<empty>`, `<exit>`, `<invoke>`, `<receive>`, `<reply>`, `<rethrow>`, `<throw>`, `<validate>`). Due to the generic tooling no extension activities, especially the BPEL4People activity was counted. This especially influences the number of basic activities in the UI processes,

All Activity Count: The aggregated count of all BPEL activities, handlers and activity-internal branches (`<elseIf>`, `<else>`, `<onMessage>`, `<onAlarm>`),

Non-Linear Activity Count: The aggregated count of all activities that enforce non-sequential control-flow. Please note that this is not the set of structured activities (`<sequence>` and `<scope>` is not included). In addition to the complex activities, handlers, and in-activity braches `<catch>`, `<catchAll>`, `<if>`, `<else>`, `<elseif>`, `<onAlarm>`, `<onMessage>` and `<onEvent>` also the basic activities `<exit>`, `<throw>` and `<rethrow>` are included,

Total Mutation Count: The aggregated count of all applyable mutations (SV, EAA, EEU, ERR, ELL, ECC, ECN, EMD, EMF, AFP, ASF, AIS, AIE, AWR, AJC, ASI, APM, APA, XMF, XMC, XMT, XTF, XER, XEE, AEL, EIU, EIN, EAP, EAN, CFA, CDE, CCO and CDC). For an explanation of these mutation operations please refer to [3].

The metrics are calculated from 2012-03-01 to 2013-05-13 without any evaluation of the date of important changes. Especially, 2012-03-01 does not indicate the start of the analyzed project.

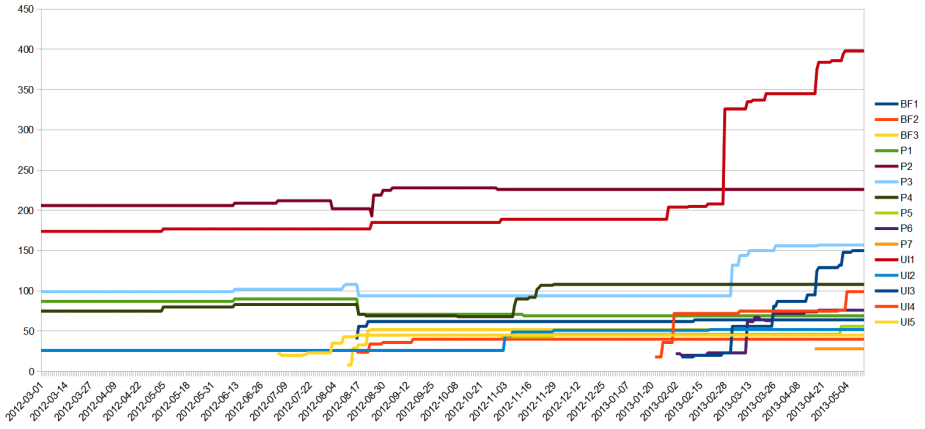

Fig. 1. Basic Activity Count for All Processes

3 Metrics of the Project

3.1 Project Overview

Within this section a general overview of the basic activity count and total mutation metrics over the whole time-span are given before looking at the processes in more details in the following section.

In figure 1 the counts for basic activities over time for all processes are shown. This figure shows the different sizes of the processes. While for a long time P2 was the largest process model it was overtaken with the latest feature release by the UI1 process, which is then the nearly twice as large by means of basic activities: The latest measurement point indicates 398 (from 174) basic activities for UI1 and 226 (from 206) basic activities for P2.

The next process is another main process: P3 (157 from initially 99 activities) followed UI3 (introduced at 2012-08-11 with 8 basic activities and now having 150 basic activities). The remaining processes are P4 (108 from 75), UI5 (108 from 22 introduced at 2012-07-05), UI4 (99 from 18 at 2013-01-22), P6 (76 from 22 at 2013-02-02), P1 (69 from 87), BF1 (64 from 40 at 2012-08-16), P5 (56 from 43 at 2012-11-01), BF3 (52 from 8 at 2012-08-11) and UI2 with the same size (52 from 26), UI5 (45 from 22 at 2012-07-05), BF2 (40 from 24 at 2012-08-16), and P7 (28 from 28 at 2013-04-17.)

The corresponding mutation metrics are illustrated in figure 2. While the absolute count of mutations is more than 10 times higher than the basic activity count, the curves are similar although the increments and decrements during changes are sometimes higher and sometimes not as high as the change in basic activity count.

In both the changes over time in basic activity count and total mutation count newly developed features and releases are clearly visible:

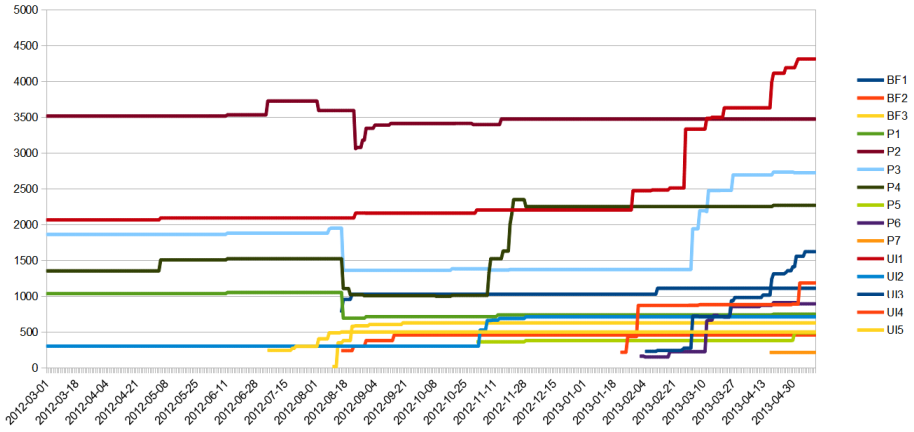

Fig. 2. Total Mutation Count for All Processes

- Starting with June 2012 the first big change was implemented that also led to a couple of refactorings for extracting out commonly functionality into the business functions in August 2012. In fact, the business function process level was introduced then (Change 1),
- In November 2012 was a small feature release that rolled out small improvements to some but not all processes (Change 2),
- In January/February 2013 development of a large new feature set was started. Development is still ongoing with the first iteration going into production at the end of May 2013, i.e. it was not finished while these measurements were taken (Change 3).

3.2 Per Process View

Within this section the diagrams for every process with all four metrics is given and discussed.

P1 (see figure 3 is one of the smallest Main Processes and the oldest one, although it has not changed much over time. Besides some small defect fixes it was only altered during Change 1: Common functionalities shared with other Main Processes were moved to the Business Function processes thereby reducing the activity count of the process. Interestingly, the number of mutations fell relatively more than the number of activities.

P2 (see figure 4) was also part of the first release and is the largest Main Process and also has the largest number of interacting partners. Interestingly, the control-flow complexity as indicated by the number of non-linear activities remains stable over time (90 non-linear activities at the start and the end of the measurement period with a small dip when business functions have been factored out). The number of all activities increased by 9.4% (from 713 to 780)

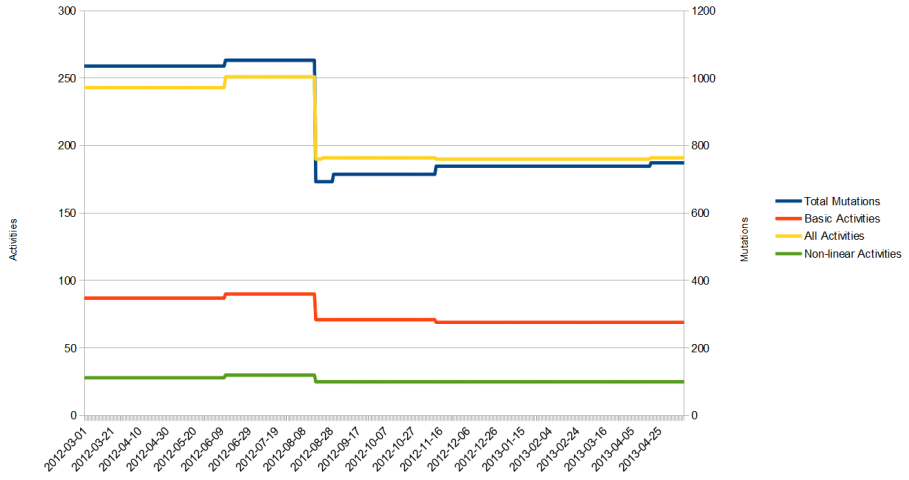

Fig. 3. All Metrics for P1

and the number of basic activities increased by 9.7% (from 206 to 226). Again, the total count of mutations spots out changes more extremely. It is noticable that at the end of the measurement period there are less mutations (3516) than at the start (3475) although the process size increased.

P3 (see figure 5) - like P1 and P2 - was also part of the first release and was affected by Change 1 and and especially Change 3. Again the total mutation count reacts more extremely than the number of activities or basic activities. The Total Mutation Count has risen by 46.2% (1864/2726) and the total number of activities by 55.4% (323/502).

P4 (see figure 6) was not part of the first release but is included in the whole measurement period. It was affected by Change 1 and especially Change 2.

P5 (see figure 7) is one of the processes that were rolled out later. It has since been extended and no functionality was extracted to new business functions. With this process all metrics except the non-linear activities rise and fall nearly proportionally: Total Mutations up by 35.2% (363/491), Basic Activities up by 30.0% (43/56) and All Activities up by 33.1% (130/173). Only the Non-linear Activities with an increase of 50% (5/10) is higher.

P6 (see figure 8) is a process that has only been introduced during the latest change which is not productive at the time of this writing. Therefore, the analyzed time-span is very short. With this process the Total Mutation Count also rises quicker than the count of All Activities.

P7 (see figure 9) is scheduled for the same release like P6 but development started even later. All changes were done in a single check-in so far so there are no observations to be made yet.

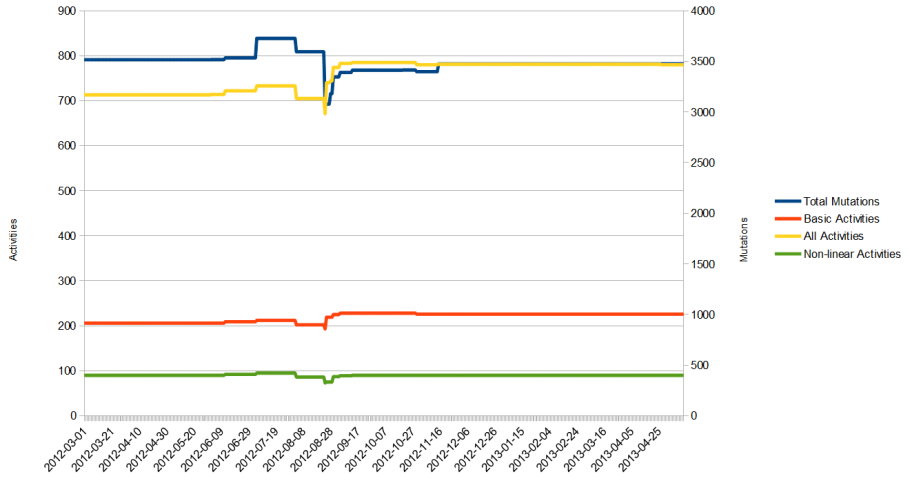

Fig. 4. All Metrics for P2

BF1 is the first instance of a Business Function process. These processes are sub-processes that contain the logic for a certain functionalities that are shared by nearly all Main Processes. All BFs are introduced with Change 1. BF1 has been upgraded with Change 3 to accommandate new functionality that was needed there. The Total Mutation Count is rising a bit sharper than the All Activity count. Due to only few commits available on this process no real facts can be conducted.

BF2 and BF3 are other Business Functions that were introduced with the first change. However, they were not modified since.

UI1 has become the largest process over time. It covers all business processes from the largest business partner type and was therefore influenced by nearly all changes. Especially Change 3 has introduced additional complexity. Like all UI processes it is to be noted that the BPEL4People extension activities are not counted due to the tooling used. This would add a considerable amount of basic activities in all UI processes.

Over time UI1 has more than doubled its size: Basic Activities increased by 128.7% (from 174 to 398), Total Mutations by 108.8% (from 2066 to 4314) and All Activities by 108.2% (from 682 to 1420). Only the Non-Linear Activities increased by only 53.8% (from 80 to 123) indicating that the control-flow complexity has increased less than the size over time.

UI2 covers all processes for another type of business partner who has far less tasks to perform that the partner of UI1. UI2 was impacted by Change 2 which doubled its size: Total Mutations by 134.2% (from 304 to 712), Basic Activities

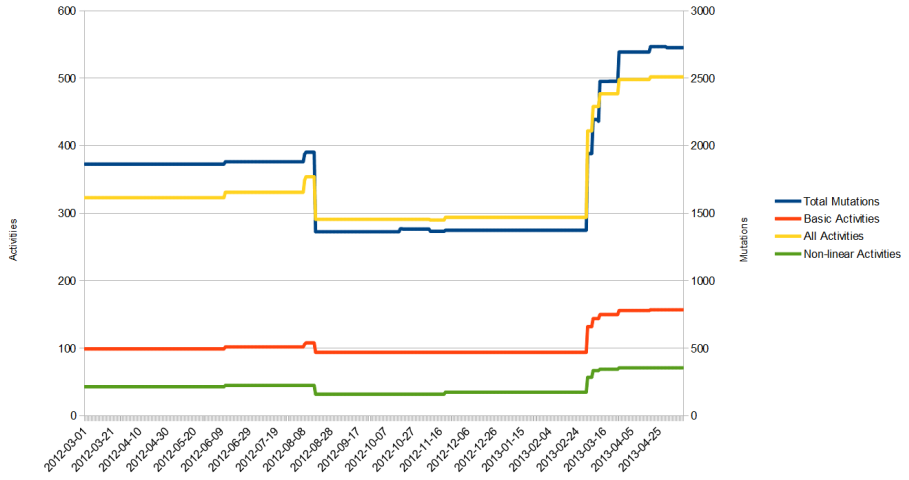

Fig. 5. All Metrics for P3

increased by 100.0% (from 26 to 52) and All Activities by 113.9% (from 101 to 216). Only Non-Linear Activities increased heavily by 176.9% (from 13 to 36).

UI3 is introduced with Change 3. The different commits reflect support for different Main Processes. Since its initial commit the process has increased by a large margin: Total Mutations by 605.7% (from 230 to 1623), Basic Activities increased by 733.3% (from 18 to 150) and All Activities by 641.8% (from 67 to 497) and Non-Linear Activities by 542.9% (from 7 to 45).

UI4 (see figure 16) is scheduled for the same release like P6 and UI4 (Change 3). Here the metrics behave like P5's metrics: They all rise more or less in parallel.

UI5 (see figure 17) behaves similarily. While being introduced with the second change all metrics are rising by similar factors.

The data of the first of every month are compiled in table 1 for Basic Activities and in table 2 for the Total Mutation Count.

4 Conclusions & Outlook

The metrics show the progress and the new features of the project quite well. Although BPEL is standardized, the standard does not offer a graphical notation. ActiveVOS offers a BPMN view and editor of the BPEL processes which hides away `<sequence>`s, `<flow>`s and other structured activities. In addition the large number of `<sequence>`s and `<scope>`s that can be seen by looking at the difference between all activities minus the basic activities minus the non-linear activities is probably a result of the editor used. However, these metrics provide a first step on judging what is a "typical" process and how could it be

Date	BF1	BF2	BF3	P1	P2	P3	P4	P5	P6	P7	UI1	UI2	UI3	UI4	UI5	Total
2012-03-01				87	206	99	75				174	26				667
2012-04-01				87	206	99	75				174	26				667
2012-05-01				87	206	99	75				174	26				667
2012-06-01				87	206	99	80				177	26				675
2012-07-01				90	209	102	83				177	26				687
2012-08-01				90	212	102	83				177	26			23	713
2012-09-01	62	36	52	71	225	94	69				185	26			45	865
2012-10-01	62	40	52	71	228	94	69				185	26			45	872
2012-11-01	62	40	52	71	226	94	68	43			189	26			45	916
2012-12-01	62	40	52	69	226	94	108	46			189	51			45	982
2013-01-01	62	40	52	69	226	94	108	46			189	51			45	982
2013-02-01	62	40	52	69	226	94	108	46			204	51		72	45	1069
2013-03-01	64	40	52	69	226	94	108	46	23		326	52	23	72	45	1240
2013-04-01	64	40	52	69	226	156	108	46	72		345	52	87	75	45	1437
2013-05-01	64	40	52	69	226	157	108	56	76	28	386	52	132	76	45	1567
2013-05-13	64	40	52	69	226	157	108	56	76	28	398	52	150	99	45	1620

Table 1. Basic Activity Count at the Start of every Month

Date	BF1	BF2	BF3	P1	P2	P3	P4	P5	P6	P7	UI1	UI2	UI3	UI4	UI5	Total
2012-03-01				1036	3516	1864	1354				2066	304				10140
2012-04-01				1036	3516	1864	1354				2066	304				10140
2012-05-01				1036	3516	1864	1354				2066	304				10140
2012-06-01				1036	3516	1864	1507				2093	304				10320
2012-07-01				1053	3534	1881	1524				2093	304				10389
2012-08-01				1053	3727	1881	1524				2093	304			300	10882
2012-09-01	1029	381	605	715	3346	1363	1007				2161	304			501	11412
2012-10-01	1029	459	629	715	3413	1363	1007				2161	304			501	11581
2012-11-01	1029	459	629	715	3398	1383	1012	363			2205	304			501	11998
2012-12-01	1029	459	629	739	3475	1374	2252	381			2205	711			501	13755
2013-01-01	1029	459	629	739	3475	1374	2252	381			2205	711			501	13755
2013-02-01	1029	459	629	739	3475	1374	2252	381			2474	711		872	501	14896
2013-03-01	1112	459	629	739	3475	1374	2252	381	226		3335	712	275	872	501	16342
2013-04-01	1112	459	629	739	3475	2694	2252	381	857		3631	712	983	883	501	19308
2013-05-01	1112	459	629	749	3475	2726	2269	491	910	215	4192	712	1412	890	501	20742
2013-05-13	1112	459	629	749	3475	2726	2269	491	894	215	4314	712	1623	1186	501	21355

Table 2. Total Mutation Count at the Start of every Month

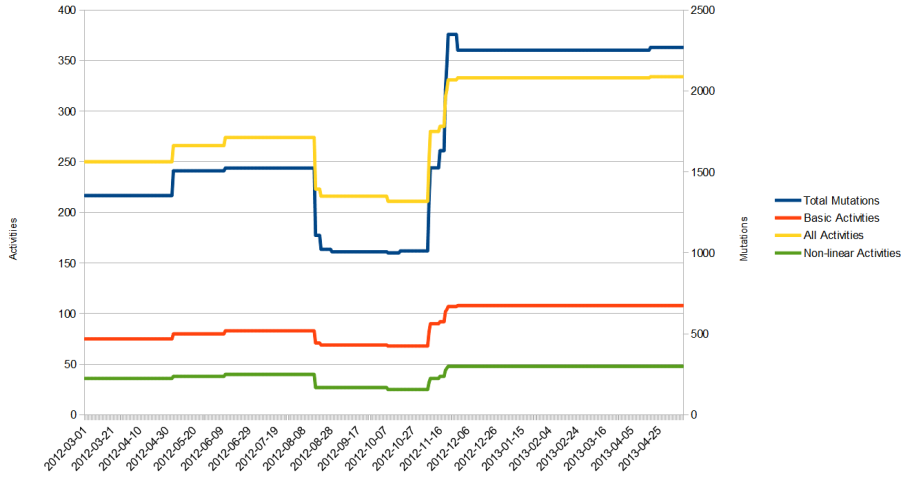

Fig. 6. All Metrics for P4

measured in a pragmatic manner. The differences for judging size by using activity counts or mutation counts is minimal. Both are not directly related as seen that in many processes the mutation count does not rise in the same way as the activity count and sometimes also decreases while the activity count increases but from a practical perspective modelers would derive the same conclusions about subjective size and complexity.

The mutation count is interesting for actually conducting mutation tests: For the given processes over 20000 process deployments and test suite runs would need to done which would – given that a test suite would on average run 1 minute, which is very optimistic – take nearly 15 days. If we assume an average execution time of 5 minutes a little more than 74 days are needed if no parallelization is available.

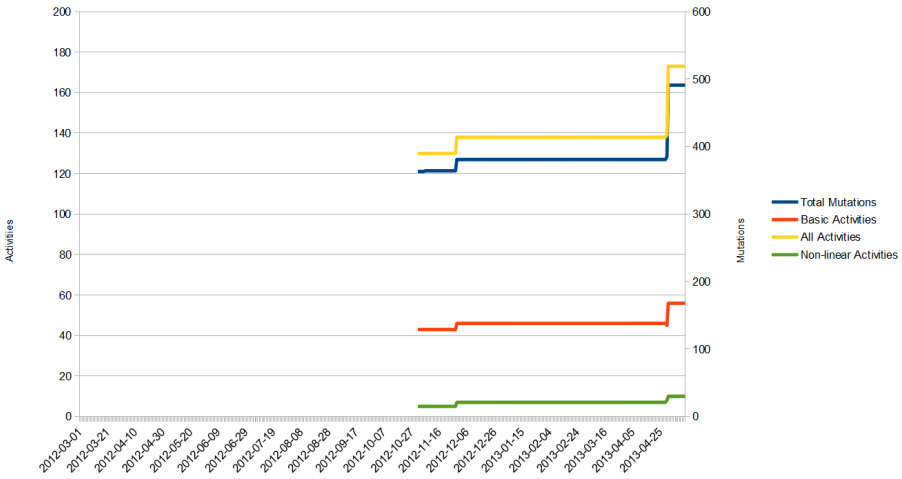

Fig. 7. All Metrics for P5

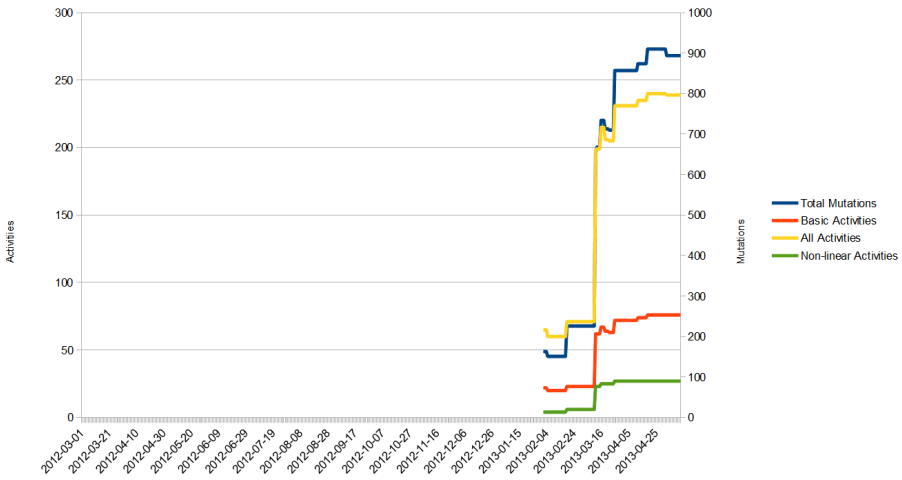

Fig. 8. All Metrics for P6

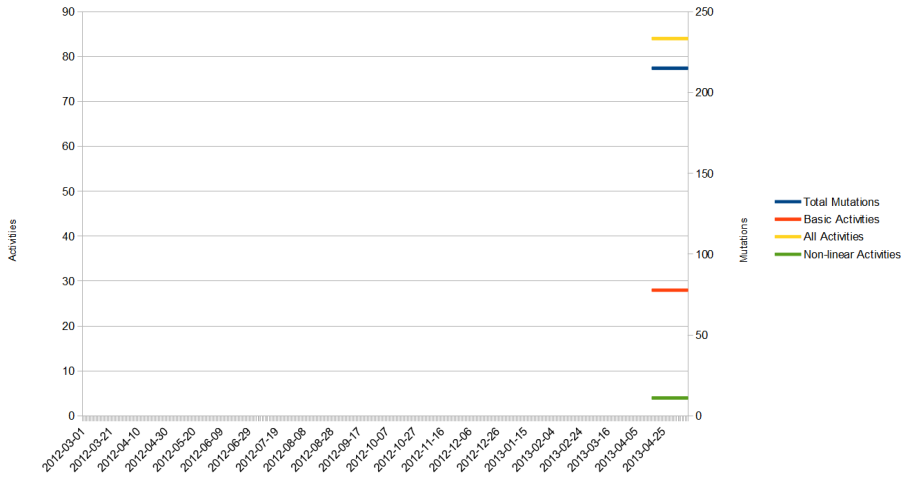

Fig. 9. All Metrics for P7

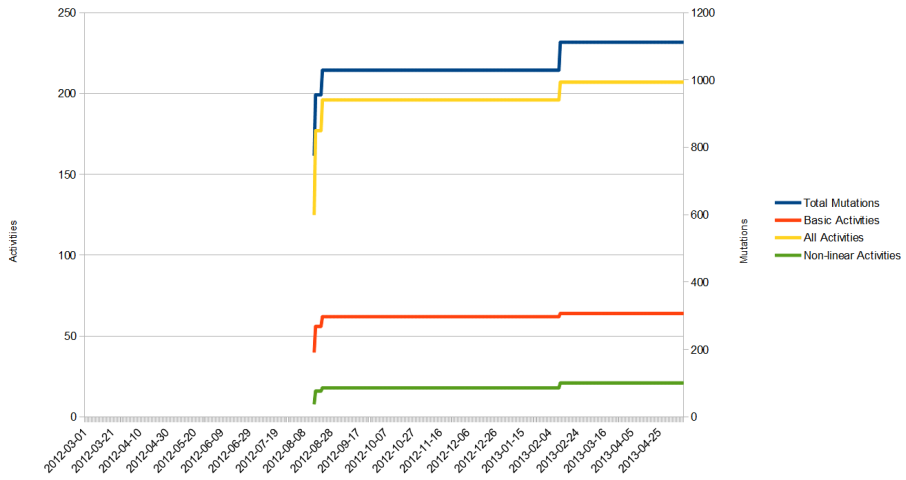

Fig. 10. All Metrics for BF1

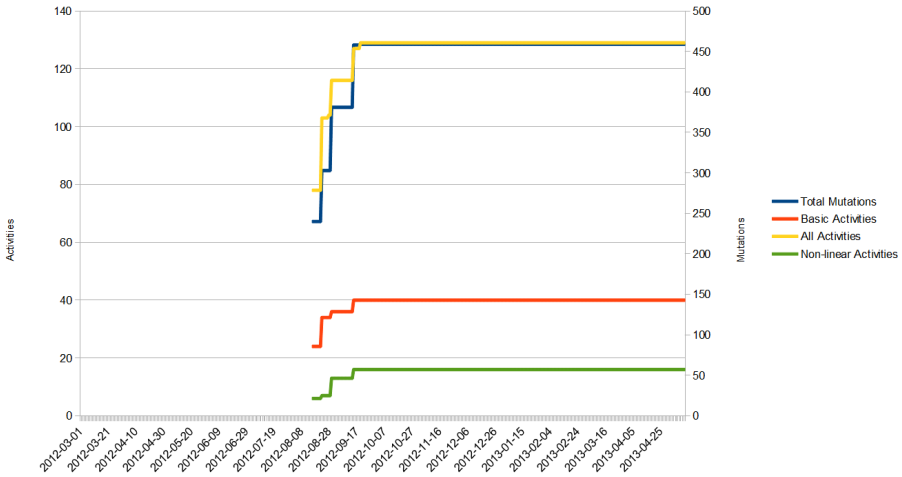

Fig. 11. All Metrics for BF2

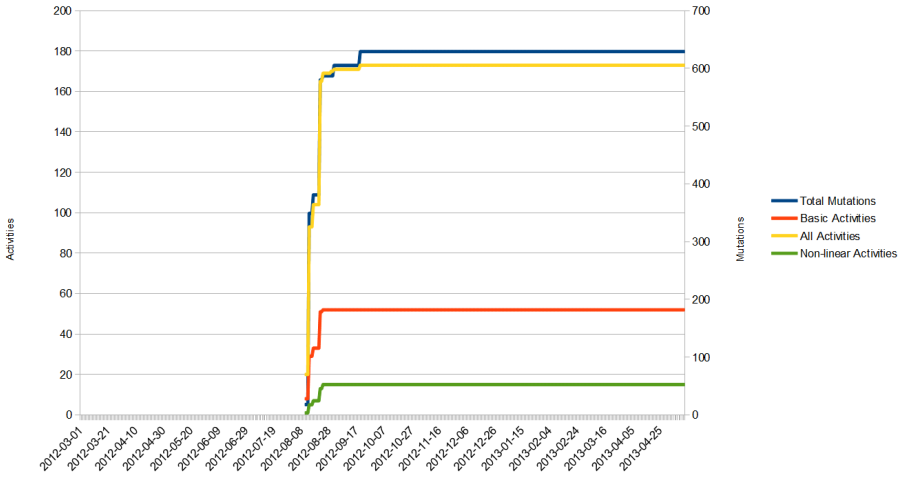

Fig. 12. All Metrics for BF3

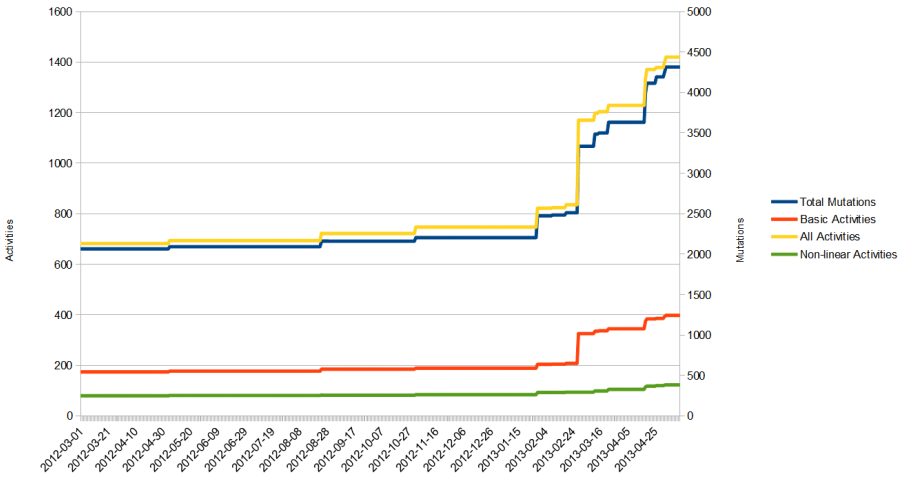

Fig. 13. All Metrics for UI1

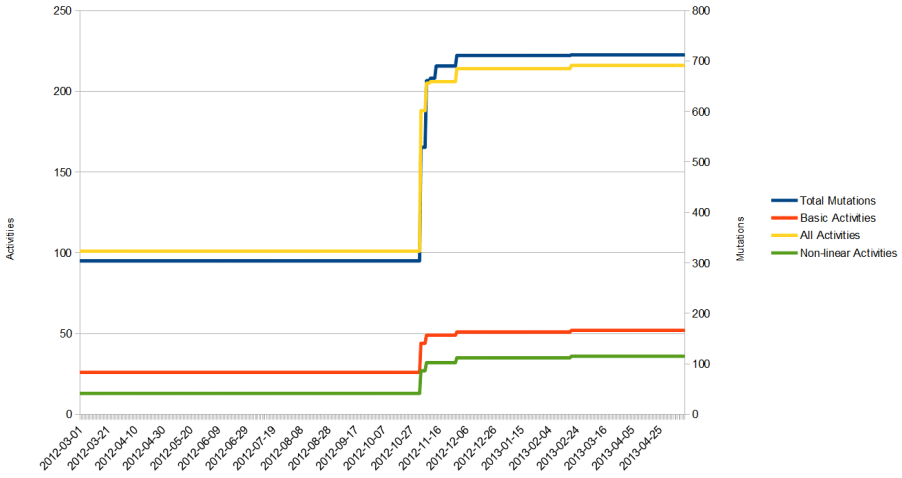

Fig. 14. All Metrics for UI2

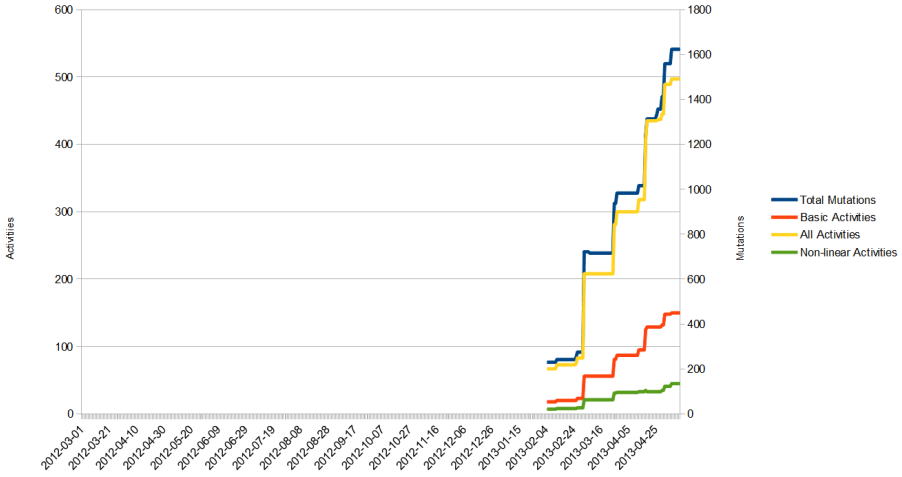

Fig. 15. All Metrics for UI3

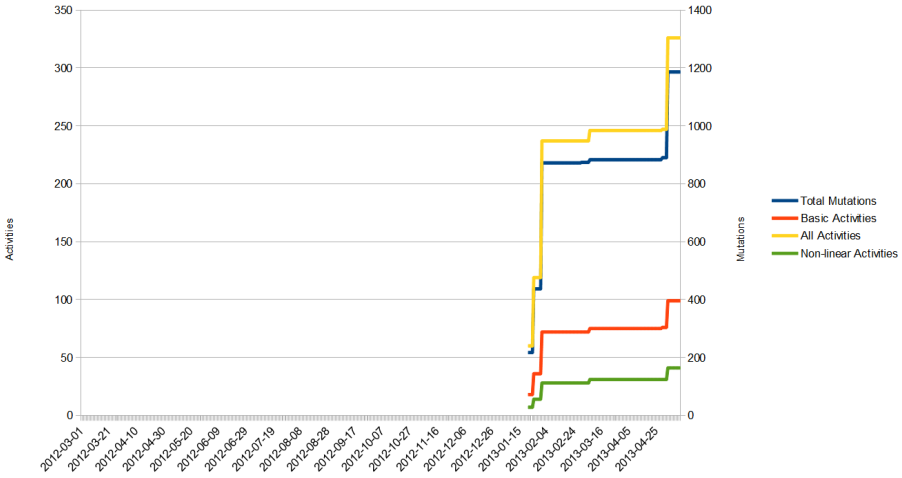

Fig. 16. All Metrics for UI4

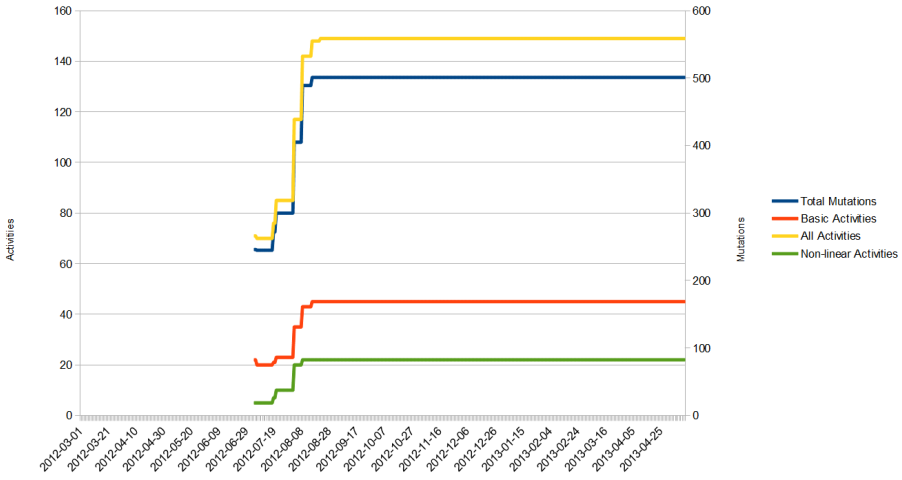

Fig. 17. All Metrics for UI5

References

1. Ashish Agrawal, Mike Amend, Manoj Das, Mark Ford, Chris Keller, Matthias Kloppmann, Dieter König, Frank Leymann, Ralf Müller, Gerhard Pfau, Karsten Plösser, Ravi Rangaswamy, Alan Rickayzen, Michael Rowley, Patrick Schmidt, Ivana Trickovic, Alex Yiu, and Matthias Zeller. WS-BPEL Extension for People (BPEL4People), Version 1.0. Technical report, IBM, 2007.
2. Jorge Cardoso. Complexity Analysis of BPEL Web Processes. *Software Process: Improvement and Practice Journal*, 12:35–49, 2006.
3. J.J. Dominguez Jimenez, A. Estero Botaro, A. Garcia Dominguez, and I. Medina Bulo. GAmera: An Automatic Mutant Generation System for WS-BPEL Compositions. In *ECOWS '09. Seventh IEEE European Conference on Web Services, 2009.*, pages 97–106, 2009.
4. Informatica. ActiveVOS Homepage. http://www.activevos.com. Last accessed 2013-05-26.
5. Diane Jordan, John Evdemon, Alexandre Alves, Assaf Arkin, Sid Askar, Charlton Barreto, Ben Bloch, Francisco Curbera, Mark Ford, Yaron Goland, Alejandro Guzar, Neelakantan Kartha, Canyang Kevin Liu, Rania Khalaf, Dieter Knig, Mike Marin, Vinkesh Mehta, Satish Thatte, Danny van der Rijn, Prasad Yendluri, and Alex Yiu. Web services business process execution language version 2.0. Oasis standard, OASIS, April 2007.
6. Daniel Lbke, Leif Singer, and Alex Salnikow. Calculating BPEL Test Coverage through Instrumentation. In *Workshop on Automated Software Testing (AST 2009), ICSE 2009*, 2009.
7. Daniel Lübke. *Test and Analysis of Service-Oriented Systems*, chapter Unit Testing BPEL Compositions. Springer, 2007.
8. Philip Mayer and Daniel Lübke. Towards a BPEL Unit Testing Framework. In *TAV-WEB '06: Proceedings of the 2006 Workshop on Testing, Analysis, and Verification of Web Services and Applications, Portland, USA*, pages 33–42, New York, NY, USA, 2006. ACM Press.
9. Universidad de Cadiz. MuBPEL Homepage. https://neptuno.uca.es/redmine/projects/sources-fm/wiki/MuBPEL. Last accessed 2013-05-26.