Reporting Scrum Project Progress to Executive Management through Metrics

Authors: Brent Barton, Ken Schwaber, Dan Rawsthorne

Contributors: Francois Beauregard, Bill McMichael, Jean McAuliffe, Victor Szalvay

January 2005

Table of Contents

Table of Figures:

Introduction

The interest in Agile software methodologies is not surprising. Agile methods are presenting an opportunity to develop software better, and this is being noticed in the business community. Scrum is particularly interesting to businesses because it focuses on Return on Investment (ROI). While the efforts of innovators and early adopters have helped us assert that Agile is better than traditional methods, improving the reporting capability to management would help. In particular, we need to be able to report project progress to executive management in a more compelling way. At a Scrum Gathering, a number of white papers were submitted and discussed. This article is a summary of those discussions and the integration of the contributions of many people. Visibility into project progress and project "health" is a consistent theme executive management desires.

Transparency into Projects

Executive Management needs transparency into all operations by viewing important indicators quickly: This is especially true of software projects. They want no surprises because software surprises are rarely pleasant. It is worth mentioning, however, that bad things do happen; executives know this and so does everyone else. It is always a surprise the first time one hears bad news. In contrast, the kind of surprise executives hate the most have significant impact <u>and</u> were known much earlier than when the executives were finally informed. The negative emotional response to the surprise is reinforced by the realization that decisions were made on faulty information and this was preventable.

There are many techniques and practices for assessing the progress and probable success of projects. Scrum provides four simple and effective artifacts for managing and monitoring project performance: Product Backlog, Product Burndown, Sprint Backlog and Sprint Burndown. Building on these, we are integrating a Functional Work Breakdown Structure and a technique for measuring Earned Business Value.

Stakeholders and executives often have particular interest in certain areas of projects. The grouping nature of a Work Breakdown Structure (WBS) affords the opportunity to present progress at a mid-level: not a single view like a burndown and not at a detail level like a backlog. By combining a WBS with a few simple, graphical reports on an executive dashboard, transparency can be attained quickly.

Executive Dashboard

The Executive Dashboard presented here is easily read, interpreted and provides the ability to reference additional material if desired (see Figure 1: Executive Dashboard).

ATM Project Dashboard

Parking Lot

26% KS	17% BB	12% DR
Login	**Withdraw Cash**	Deposit Check
(5)	(4)	(3)
(100%)	(40%)	(32%)
January 2005	February 2005	March 2005

25% KS	17% DR	2% BB
Business Support Activities	Transfer Funds	Buy Stamps
(6)	(7)	(2)
(17%)	(0%)	(0%)
Planned Completion	March 2005	March 2005

LEGEND

BVI   Owner

Feature Group, Use Case Package or Use Case	☐ Completed
# features or stories	☐ Attention
	☐ In Process
% complete	☐ Not Started
Progress Bar	
Planned Completion	

Business Value

Product Burndown

Estimated Completion
May 3, 2005

Work Breakdown Structure

Space for:
 Links
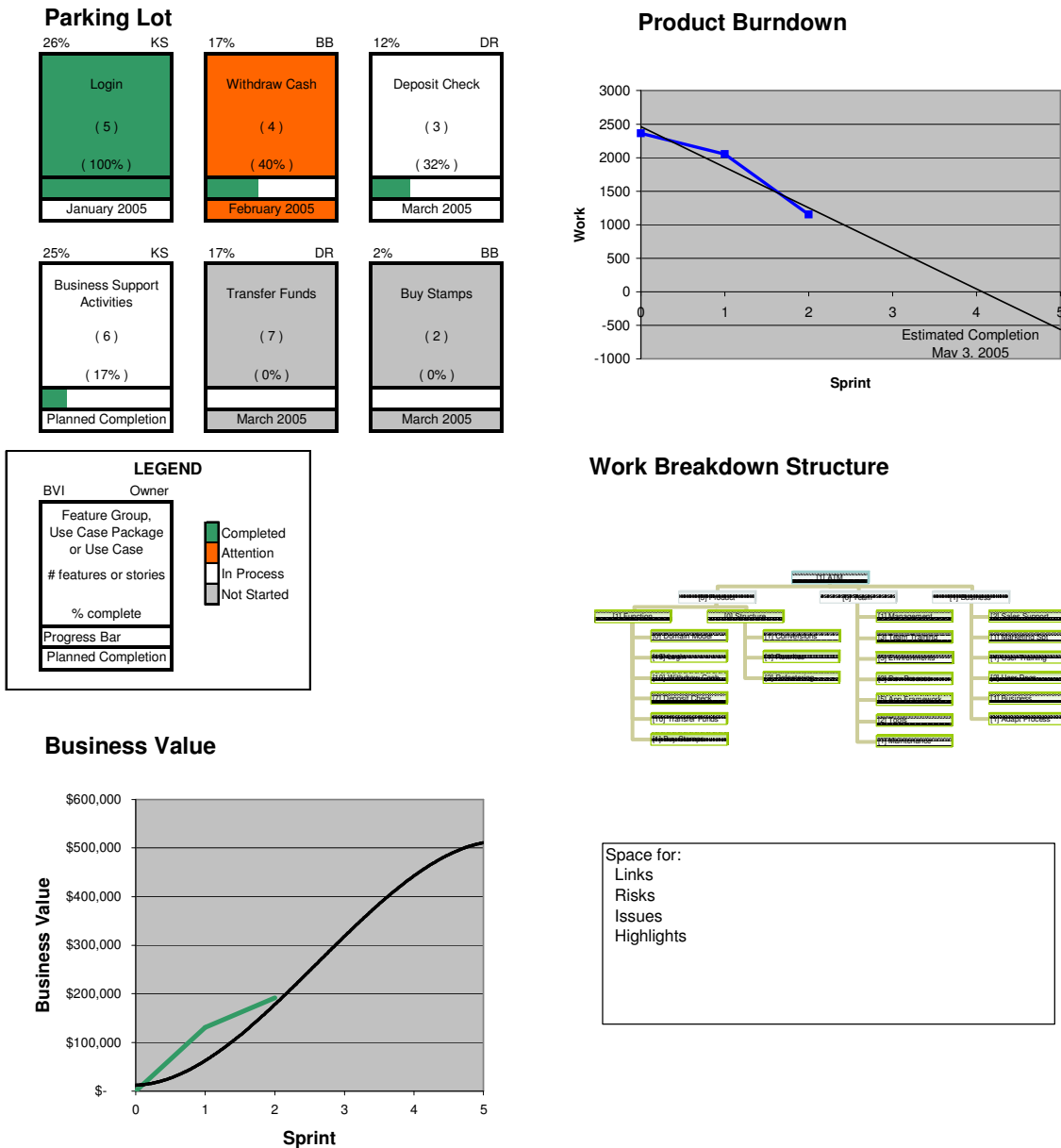 Risks
 Issues
 Highlights

Figure 1: Executive Dashboard

The contents of this dashboard report include:

- Parking Lot: This is a pictorial that statuses groups of features or use cases. This has been adopted from reports found in Feature Driven Development (FDD). With the addition of a Business Value Index (described later), one can see the progress and value of this area to the business. At a glance, the colors show where progress is made, areas of concern are and items not started. The BVI indicated what percentage of the total value of the project this activity is worth;

and the owner's initials describe who is responsible for the activity. The legend is included (see Figure 2: Parking Lot).

26% KS	17% BB	12% DR
Login (5) (100%) — January 2005	Withdraw Cash (4) (40%) — February 2005	Deposit Check (3) (32%) — March 2005

25% KS	17% DR	2% BB
Business Support Activities (6) (17%) — Planned Completion	Transfer Funds (7) (0%) — March 2005	Buy Stamps (2) (0%) — March 2005

LEGEND

BVI   Owner

Feature Group, Use Case Package or Use Case

features or stories

% complete

Progress Bar

Planned Completion

- Completed
- Attention
- In Process
- Not Started

Figure 2: Parking Lot

- Product Burndown: The burndown in work budgeted and planned compared as decreased by work completed across time. Based upon this, an estimated completion date can be determined as the trend line crosses the x-axis (see Figure 3: Product Burndown).
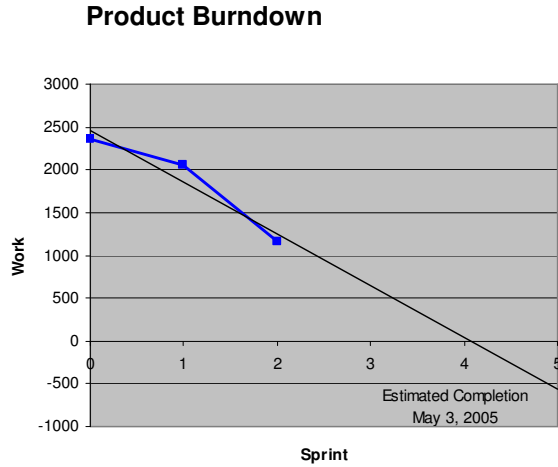
Product Burndown

Figure 3: Product Burndown

- Earned Business Value Graph: This presents the Business Value earned compared to the Planned Business Value. Variance can be quickly estimated from the graph to assess the correct prioritization and progress of the project Figure 4: Business Value Burnup).
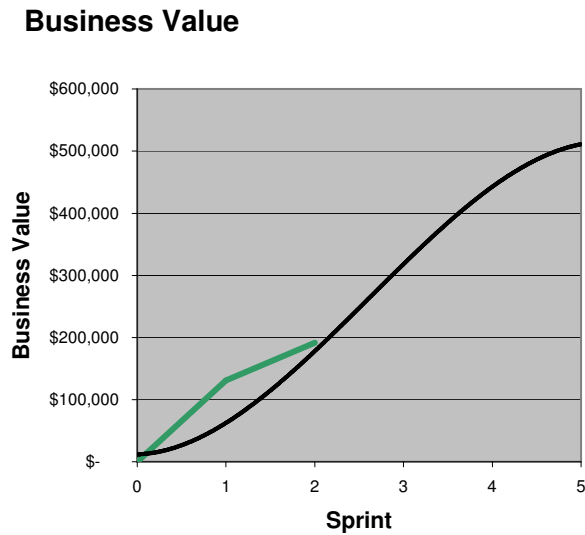
Business Value

Figure 4: Business Value Burnup

- Functional Work Breakdown Structure: This visual representation provides a concise, high-level presentation of the project work items (see Figure 5: Functional Work Breakdown Structure).

- Space for links, highlights, issues and risks. Every project and customer has its own specific needs. This space is intended for a few bullet points.

Work Breakdown Structure

Dan Rawsthorne introduced a functional Work Breakdown Structure (WBS) which provides us a structure for reporting key areas within a project and also measuring Earned Business Value[1]. A Work Breakdown Structure provides "'A deliverable-oriented grouping of project elements which organizes and defines the total scope of the project.' (Source: PMI: PMBOK)"

Many think of Gantt charts and Microsoft Project Plans when they hear the term Work Breakdown Structure. The visually appealing format pictured below allows anyone to quickly see the salient work required to accomplish the project (Project at a Glance). This sample software project's WBS looks like the following, representing a fictitious ATM development project (see Figure 5: Functional Work Breakdown Structure).
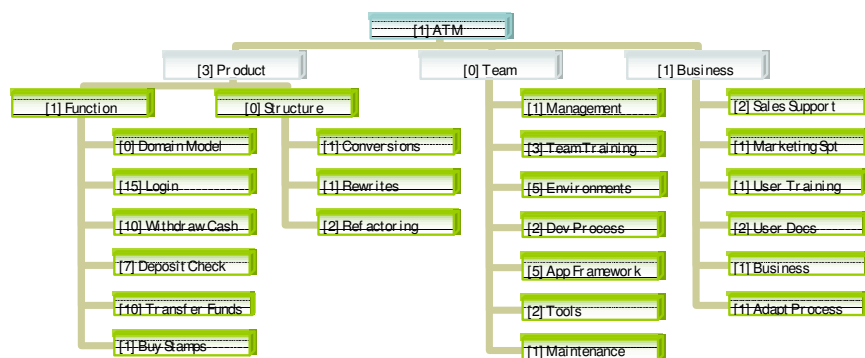

Figure 5: Functional Work Breakdown Structure

The bottom nodes in the Functional leg of this WBS represent use cases. A use case is not required but I believe scenarios of use cases and stories align well and help produce useful software. Using other forms of requirements does not invalidate this structure.

The numbers you see in the nodes are used to calculate business value metrics. They are additive values that provide relative weights of importance to their parent in the WBS. Those that have value "0" provide no value towards the value of their parent. Note that it is possible for a parent to have no value, but its children to have value within the parent. This is because while the overall area provides no benefit to the project, the manager of that area wants to see how his "pieces" are coming along relative to each other.

Earned Business Value

In order to represent the Earned Business Value (EBV) of a project and its components, the additive weights need to be assigned to the nodes of the WBS, and an additive value of the whole project needs to be determined. Total Business Value of the project is determined by some ROI calculation or equivalent.

The Business Value of a node becomes earned only when the lowest level items are done. In Scrum terms this means it is an "increment of potentially shippable functionality."

---

[1] http://netobjectives.com/resources/downloads/ManagingTheWork.pdf

Thus, only items of direct business value, such as functionality, sales, or marketing should have "rolled up" business value other than zero. The other items are the cost of doing business. They have value, but it is not "business value". By calling them "orphans" this indicates that they need to be adopted by items that do have value (Note: This is useful because it addresses total cost, not just cost-per-feature of a project and makes visible the cost of doing business in software. Also, the software team is reminded of the difference between important work and business value of the output).

The minute, gory, details of addressing these weights would mean repeating Dan's entire paper here so this is a summary. Please read it if you have any questions. Attached with this is a spreadsheet where the calculations are already in place.

In order to apply Business Value (BV) to a project, we need to calculate the Business Value Indices. The Business Value Index (BVI) of the entire ATM project equals 1 (you get 100% of the value if you do it all). As you move down the WBS, the BVI gets portioned out to the nodes based on their relative weights. To calculate the BVI of nodes when we go down the tree, we are apportioning the BVI of a parent node to its children based on their relative weights.

This seems complex, but the pattern is easy once you understand it. Let's look at the top-level nodes. Because the functional leg has a weight of 3, team leg has a weight of 0, and the business leg has a weight of 1, the sum of the additive weights of this level is 3+0+1=4. Thus, the relative weight of the functional leg is 3/4 and that of the business leg 1/4; this was obtained by dividing the weight of the leg by the sum of the weights within its parent node. Since the BVI of their parent is 1 (the whole thing), their BVIs are 3/4 and 1/4, respectively.

Using the same method, the BVI of the login use case is 1 * 3/4 * 1 * 15/43 = 45/172. If the Total Business Value of the project is $500,000, then the Earned Business Value (EBV) realized by completing the Login Use case would be $130,814.

The Underlying Data

Managing and reporting effectively is a lot of work. The validity of the reports is only as good as the validity of the data. Figure 6: WBS in Product Backlog Format captures the WBS and calculates the BVI for each level. Note that each use case has been broken into stories (scenarios) and the weights and BVI have been calculated as well. For each item marked as done, the EBV is calculated. Notice the bottom row is done but there is no EBV because this represents a cost of doing business.

Area	Sub-Level	Use Cases	Value	BVI	Stories or Features	Value	BVI	Estimate	Done	Sprint	EBV
Product	Function	Login	15	26.2%	determine postconditions	0	0%	8	1	1	$ -
Product	Function	Login			Determine Main Success Scenario	0	0%	16	1	1	$ -
Product	Function	Login			Code up Main Success Scenario	10	20%	80	1	1	$ 100,626
Product	Function	Login			Extensions	0	0%	30	1	1	$ -
Product	Function	Login			Code Up "3 Strikes and You're Out"	3	6%	16	1	1	$ 30,188
Product	Function	Withdraw Cash	10	17.4%	determine postconditions	0	0%	12	1	2	$ -
Product	Function	Withdraw Cash			Determine Main Success Scenario	0	0%	20	1	2	$ -
Product	Function	Withdraw Cash			Code up Main Success Scenario	7	12%	120	1	2	$ 61,047
Product	Function	Withdraw Cash			Extensions	0	0%	30	0	2	$ -
Product	Function	Withdraw Cash			Code Up "Quick Cash Options"	3	5%	20	0	2	$ -
Product	Function	Deposit Check	7	12.2%	determine postconditions	0	0%	20	1	2	$ -
Product	Function	Deposit Check			Determine Main Success Scenario	0	0%	20	1	2	$ -
Product	Function	Deposit Check			Code up Main Success Scenario	10	10%	100	0	2	$ -
Product	Function	Deposit Check			Extensions	0	0%	30	1	2	$ -
Product	Function	Deposit Check			Code Up "Deposit Foreign Currency"	2	2%	50	0	2	$ -
Product	Function	Transfer Funds	10	17.4%				300	0	3	$ -
Product	Function	Buy Stamps	1	1.7%				100	0	3	$ -
Product	Structure	Conversions		0.0%				200	0		
Product	Structure	Rewrites		0.0%				300	0		
Product	Structure	Refactoring		0.0%				250	0		
Team	Team	Management		0.0%				120	0		
Team	Team	Team Training		0.0%				40	1	1	

Figure 6: WBS in Product Backlog Format

So, how does this information produce a dashboard?

Putting the Data into the Report

The Product Burndown is automatic, using the columns, Estimate, Done and Sprint. Rather than complete automation, the auto-filter feature in Excel is applied for each Sprint and this can then be easily tabulated into this table (see Figure 7: Product Burndown). This is found in the header section of the Product Backlog in the attached spreadsheet. Using the subtotal function in Excel provides quick calculations from filtered data. Applying no filter calculates the Work Left before the start of Sprint 1. Sprint 1 was found by using a custom filter where Sprint "does not equal 1." Sprint 2 was found by applying the filter where Sprint "is greater than 1." As long as any items that are dropped form an active Sprint is reflected in the Product Backlog as planned for the next Sprint (unless the Product Owner changes that to a Sprint further out).

Sprint	Work Left
0	2392
1	2052
2	1152
3	
4	
5	

Figure 7: Product Burndown

Creating the Parking Lot

The Parking Lot is created using the filters as well. For the "Deposit Check" use case, filter on Deposit Check in the use case column. The number of stories equals 5 so this will go in that middle of the diagram (see Figure 8: Deposit Check Progress Indicator). The total estimated time to completion is 220 and the total estimate of time where "Done" is filtered to 1 is 70. Thus, the percentage complete is 70/220 ~32%. Using the elements in the "Parking Lot Construction" tab in the spreadsheet, you can copy and paste the progress bar, so it graphically approximates this completion value. Since Scrum uses forward looking estimates, it is possible for a use case or feature group to show negative progress compared to a previous report as new information causes estimates to increase. The date is formatted in month and year but can be reformatted to reflect the lengths of iterations.
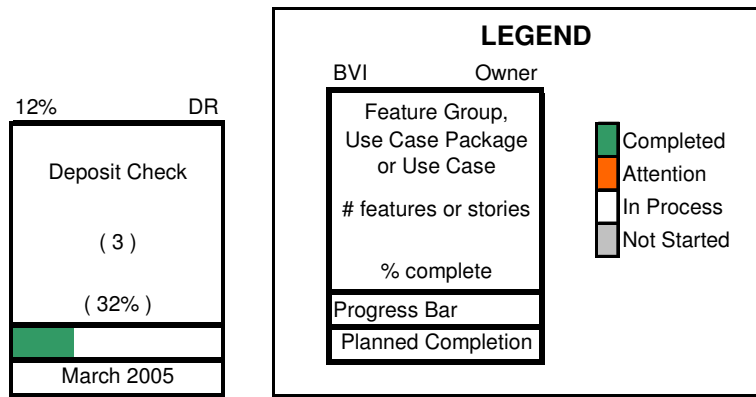
Figure 8: Deposit Check Progress Indicator

Earned Business Value

Earned Business Value is easily calculated by filtering on the Done field for a value of 1. Then filter on the Sprints up to and including the one being calculated using the custom filter "is less than or equal to." This yields the data in the EBV column for each Sprint (see Figure 9: Earned Business Value Data). This is located in the "EBV" tab in the accompanying spreadsheet. Note the Planned Business Value is calculated initially. Value is typically realized using an 'S' curve. The sample here has only 5 data points so the smoothing feature was used for the Planned Business Value. If the project is highly emergent (little planning beyond the current Sprint) the planned business value will only be one row ahead of the earned business value.

Sprint	Planned Business Value	Earned Business Value
0	$ -	$ -
1	$ 100,000	$ 130,814
2	$ 150,000	$ 195,767
3	$ 300,000	
4	$ 475,000	
5	$ 500,000	

Note:
This column represents skewed data because of graph smoothing. Because there are only 5 data points, a standard S-shaped Business Value is not represented well without smoothing

Figure 9: Earned Business Value Data

Conclusion

By taking advantage of the visibility that Agile methodologies provide, we can deliver meaningful information all the way up to Executive Management using graphical representations. The addition of a functional work breakdown structure provides the ability to view the project at multiple levels of granularity.

Notes

Many other metrics can be derived from the core data, including early warning indicators and schedule variances. Care should be taken not to violate the principles of Agility by imposing too much data analysis or tracking obscure data. The KISS (Keep It Simple, Simon) principle applies well here.

How weights are applied so they are additive can be debated or can be simple. This should be coordinated with the business and financial people to use their calculators to help derive this.