

# **Reporting Scrum Project Progress to Executive Management through Metrics**

Brent Barton, Ken Schwaber, Dan Rawsthorne

Contributors: Francois Beaugard, Bill McMichael, Jean McAuliffe, Victor Szalvay  
Scrum Alliance, 2005

## **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 of interest partly because of its ROI focus and quick implementation. While the efforts of innovators and early adopters have helped us assert that Agile is better than traditional methods, improving the reporting capability would help. Even better would be able to report project progress to executive management in a more compelling way. At a Scrum Gathering, white papers were submitted and discussed. This 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 in software a surprise is rarely a pleasant one. 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 and were known much earlier than when they 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, transparency can be attained quickly with a few simple, graphical reports on an executive dashboard

# Executive Dashboard

The Executive Dashboard presented here is easily read, interpreted and provides the ability to reference additional material if desired (see Figure 2: Executive Dashboard).

## ATM Project Dashboard

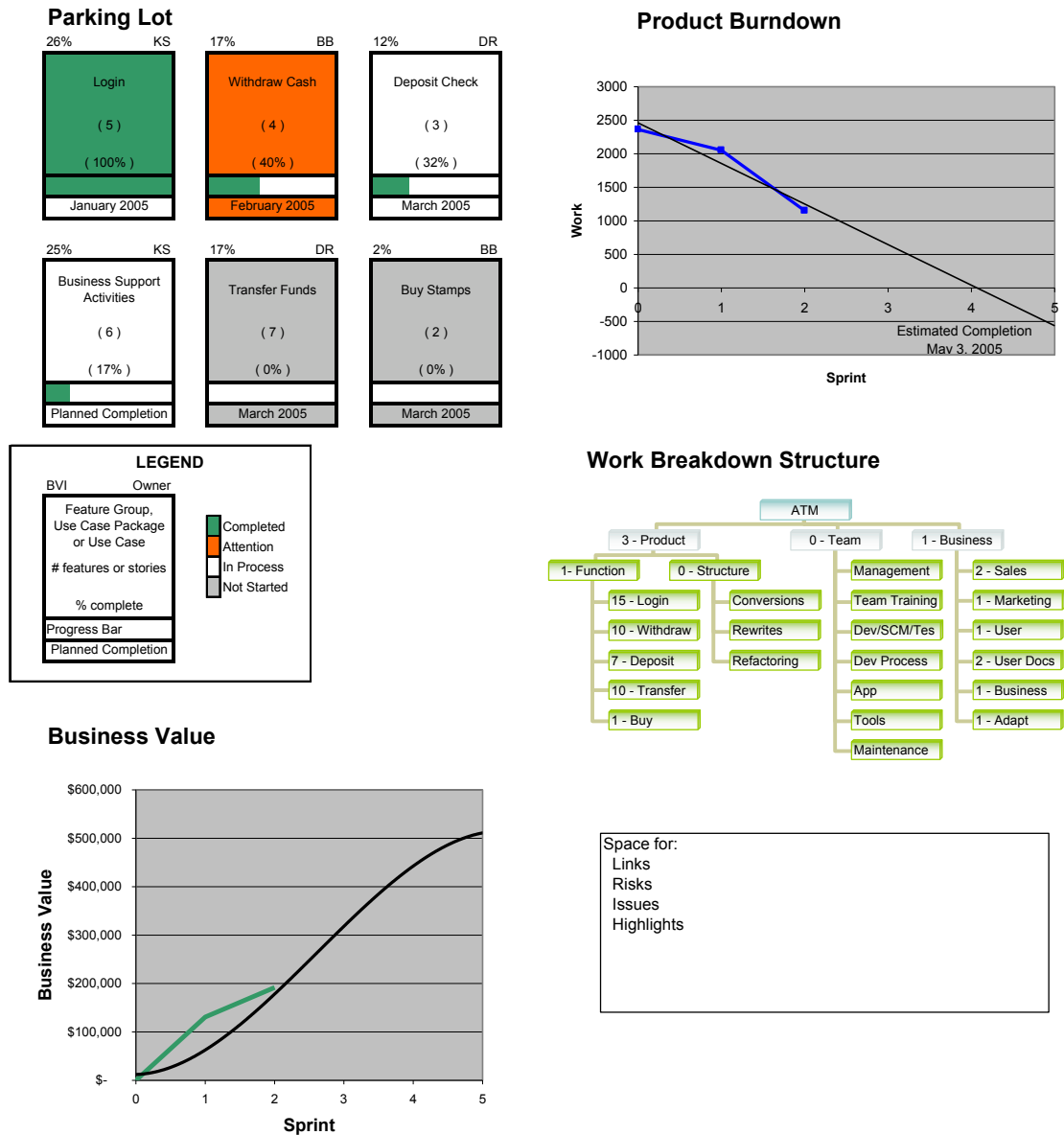
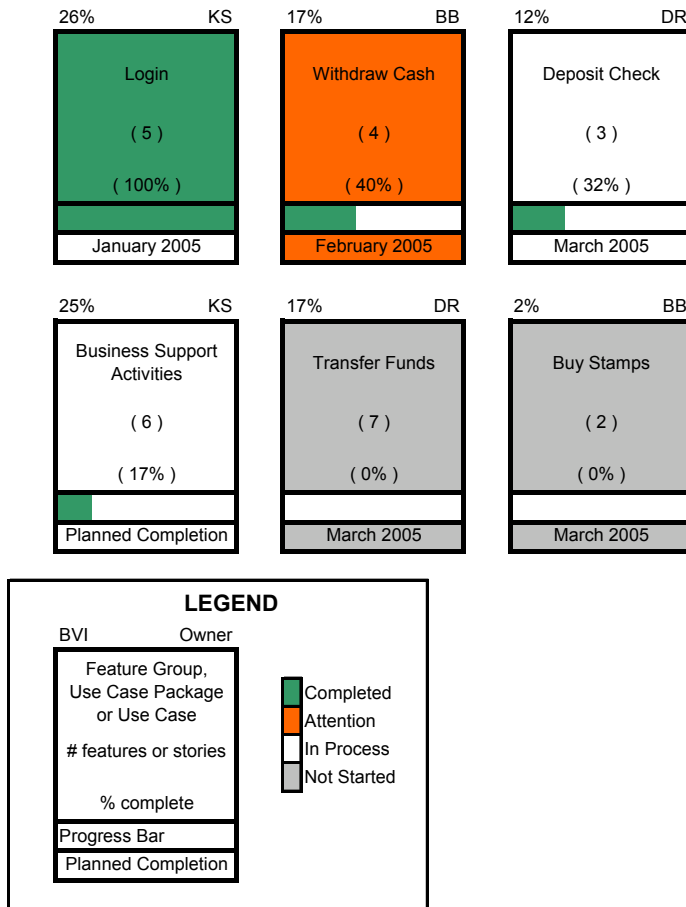


Figure 1: Executive Dashboard

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 represents the total value of the project and the owner's initials describe who is responsible for the groups. The legend is included (see **Figure 3: Parking Lot**).



**Figure 2: Parking Lot**

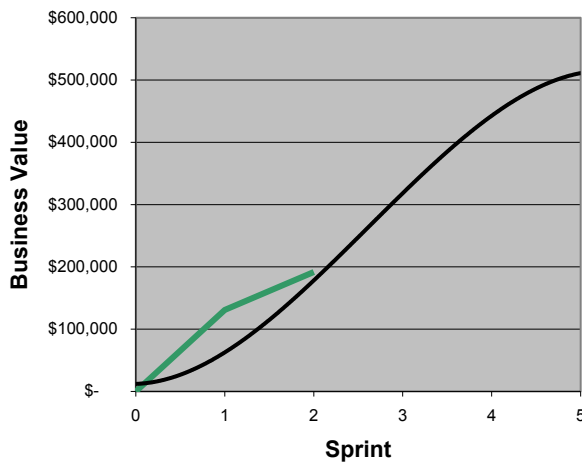
**Product Burndown:** The burndown in work budgeted and planned compared as decreased by work completed across time. Based upon this, an estimated completion date can be determined as the trend line crosses the x-axis (see **Figure 4: Product Burndown**).

bbb

**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 (see **Figure 5: Business Value Burnup**).

## Business Value



**Figure 4: Business Value Burnup**

Graphical Work Breakdown Structure: This visual representation provides a concise, high-level presentation of the project work items (see Figure 6: Functional Work Breakdown Structure).

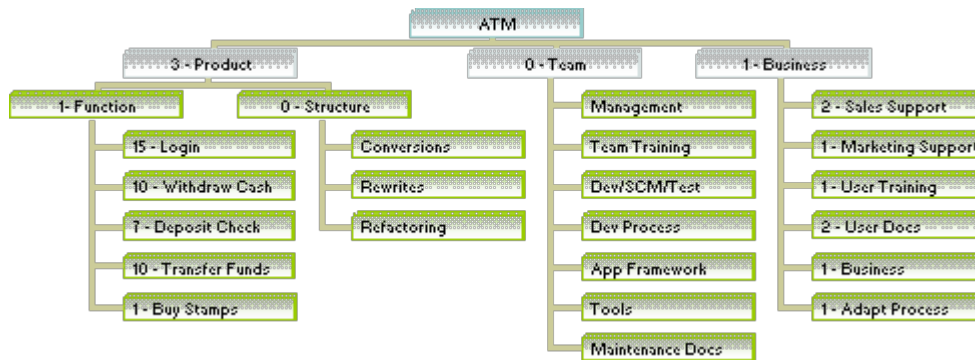
Space for links, highlights, issues and risks. Every project and customer has its own specific needs. This space is intended for a few bullet points.

### *Work Breakdown Structure*

Dan Rawsthorne introduced a functional Work Breakdown Structure which provides us a structure for reporting key areas within a project and also measuring Earned Business Value. A Work Breakdown Structure provides “‘A deliverable-oriented grouping of project elements which organizes and defines the total scope of the project.’ [87]

b

Many think of Gantt charts and Microsoft Project Plans when they hear the term Work Breakdown Structure. This visually appealing format 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 6: Functional Work Breakdown Structure).



**Figure 5: Functional Work Breakdown Structure**

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

Notice the numbers in the nodes that represent tangible things that can be valued. The other items are necessary to deliver the required results but do not have direct business value.

### ***Earned Business Value***

In order to represent the Earned Business Value (EBV) of a project and its components, an additive weight needs to be assigned. Total Business Value is determined by some ROI calculation or equivalent. Business Value becomes earned only when the items are done. In Scrum terms this means it is an “increment of potentially shippable functionality.” Thus, only items of direct business value, such as functionality and training should be assigned weights other than zero. The other items are the cost of doing business. By calling them “orphans” 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 the difference between important work and business value of the output).

Detailed calculation of EBV is was published in the Agile 2006 proceedings . Here, only a brief overview is provided for one calculation [86]. In order to apply Business Value (BV) to a project, we need to calculate the Business Value Indexes. The Business Value Index (BVI) of the entire ATM project equals 1. For each level in the WBS, the index is 1: This is an intermediate value that will be used to calculate the BVI. To calculate the BVI of subsequent levels, you must also multiply the BVI of the all nodes above the level you are considering.

The next part is a bit more complex but the pattern is easy once you understand it. Remember, the BVI of the next level is  $1*1=1$ , the index of the Project times the index of the next lower level. Because the functional leg has a weight of 3 and the business leg has a weight of 1, the sum of the additive weights of this level is  $3+1=4$ . Thus, the Index

of the functional leg is 3/4 and the business leg 1/4: This was obtained by dividing the weight of the leg by the sum of the weights on this leg. This is why additive weights must be used. So in order to calculate the BVI, multiply the index of the target leg by the index of all the legs above it. Since the only leg above the functional leg is the ATM project,  $3/4 * 1 = 3/4$ . the weight of the leg by the sum of the weights of all legs. Thus the BVI for the functional leg is 3/4 and for the business leg is 1/4.

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 7: 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

Sprint	Work Left
0	2362
1	2052
2	1152
3	
4	
5	

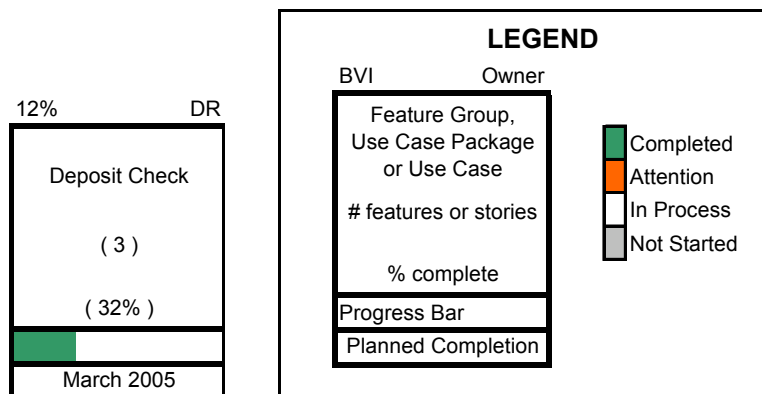
The Product Burndown is automatic, using the columns, Estimate, Done and Sprint. Rather than complete automation, the auto-filter feature in Excel is applied for each Sprint and this can then be easily

Figure 7: Product

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 “is not equal to 1”. Sprint 2 was found by applying the filter where Sprint “is greater than 1.” As long as any items that are dropped from 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).

### *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  $70/220 \sim 32\%$ . 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.

## Conclusions

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.

Sprint	Planned Business Value	Earned Business Value
0	\$ -	\$ -
1	\$ 100,000	\$ 130,814
2	\$ 150,000	\$ 191,861
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

## Notes

Many other metrics can be derived from the core data, including early warning indicators and schedule variances.

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.

**Figure 9: Earned Business Value Data**