goto; conference

Randomized Branch Sampling (RBS): Size software projects without wasting time analyzing each user story

Dimitar Bakardzhiev

goto; conference

Click ‘engage’ to rate sessions and ask questions

Join the conversation #gotocph

How big is our project?

Sizing estimates the **probable size** of a piece of software while effort estimation estimates the **effort** needed to build it.

Agile sizing techniques

measure User Stories

T-Shirt sizes (Small, Medium, Large and so on)

http://www.mountaingoatsoftware.com/blog estimating-with-tee-shirt-sizes

Story points (Fibonacci numbers or Exponential scale)

http://www.mountaingoatsoftware.com/blog/dont-equate-story-points-to-hours

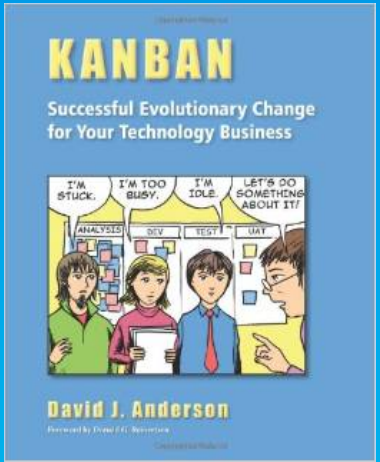
Story points are about effort.

But…software sizing is different from software effort estimation!

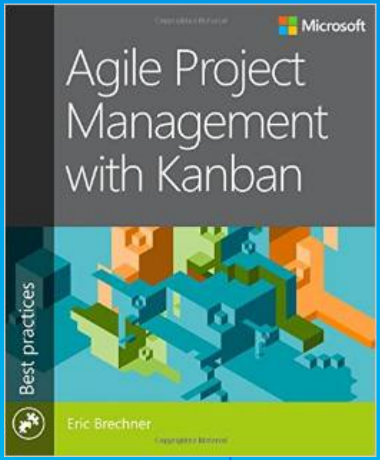
http://www.mountaingoatsoftware.com/blog/story-points-are-still-about-effort

Kanban project sizing? Count!

Number of user stories, features, use cases


KANBAN
Successful Evolutionary Change for Your Technology Business
I'M STUCK.
I'M TOO BUSY.
I'M IDLE.
LET'S DO SOMETHING ABOUT IT!
ANALYSIS
DEV
TEST
UAT
David J. Anderson

Number of tasks



Project size is the total of "work items suitable for the development organization."

Example sizing

- We have identified 16 epics in our project
- We have identified that those 16 epics contain 102 **user stories** in total
- We have analyzed and sized every single one of those 102 user stories and arrived at a total number of 896 **story points** for our project

This practice is time consuming and probably great part of this effort will be a pure waste!

How can we estimate the total number of story points or tasks for a project without prior identification, analysis and sizing of every single user story?

Randomized Branch Sampling

Raymond J. Jessen
1910–2003

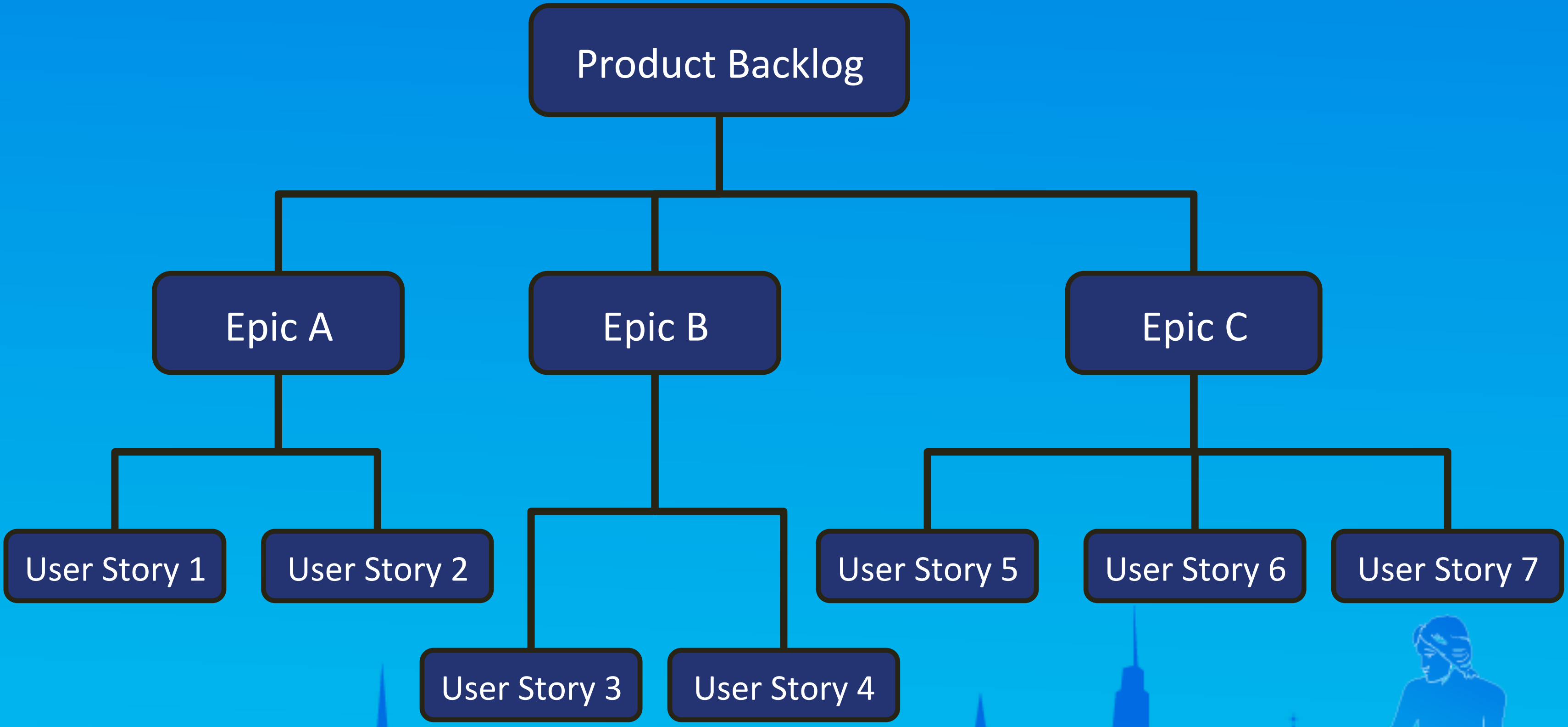
The technique was designed to efficiently estimate the total number of fruit found in the canopy of a tree while only having to count the fruit on select branches. RBS is a method for sampling tree branches which does not require prior identification of all branches, and provides the sampler with unbiased tree level estimates.

Randomized branch sampling (RBS)

- A multi-stage unequal probability sampling method which doesn't require prior identification of all branches in the crown, and provides the sampler with unbiased tree level estimates
- Designed to efficiently estimate the total number of fruit found in the canopy of a tree while only having to count the fruit on select branches
- A tree level estimate is derived by combining the number of fruit from the terminal branch and the associated probability with which that particular branch was selected

goto; conference

Product backlog as a branching system



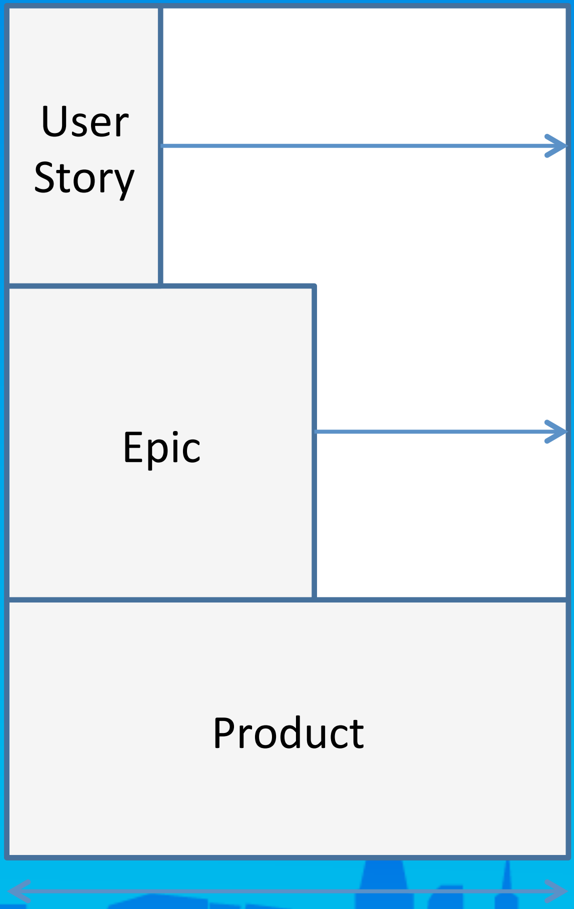
Horvitz-Thompson estimator

$$\hat{X} = \frac{x_i}{Q_i}$$

Unconditional selection probability (Q_i)

$$Q_i = \prod_{k=1}^{i} q_k = q_{product\ backlog} q_{epic} q_{story} = q_{epic} q_{story}$$

Applications of RBS to project sizing



Total size per section level

The user story rectangle represents the estimated size of a randomly sampled user story. The size of that user story is expanded to an estimated total project size by dividing that size by its selection probabilities which is indicated here by the arrows. The selection probabilities assigned to epics and user stories are arbitrary.

Why RBS works for sizing software development projects?

The assumption behind using RBS for software development is that project size depends on the context – the customer, the people developing the product and the methodology they use for managing the requirements, breaking down the product into stories and sizing a story.

It doesn’t matter what the methodology is – Planning Poker, Product Sashimi, Behavior Driven Development etc. What is important is that the methodology be cohesive, explicit and to be consistently applied during project execution when we slice the requirements into user stories.

Applications of Randomized branch Sampling (RBS)

RBS for checking team’s consistency

- If the training/coaching in a sizing methodology was successful?
- Can we trust our historical “story points” data?

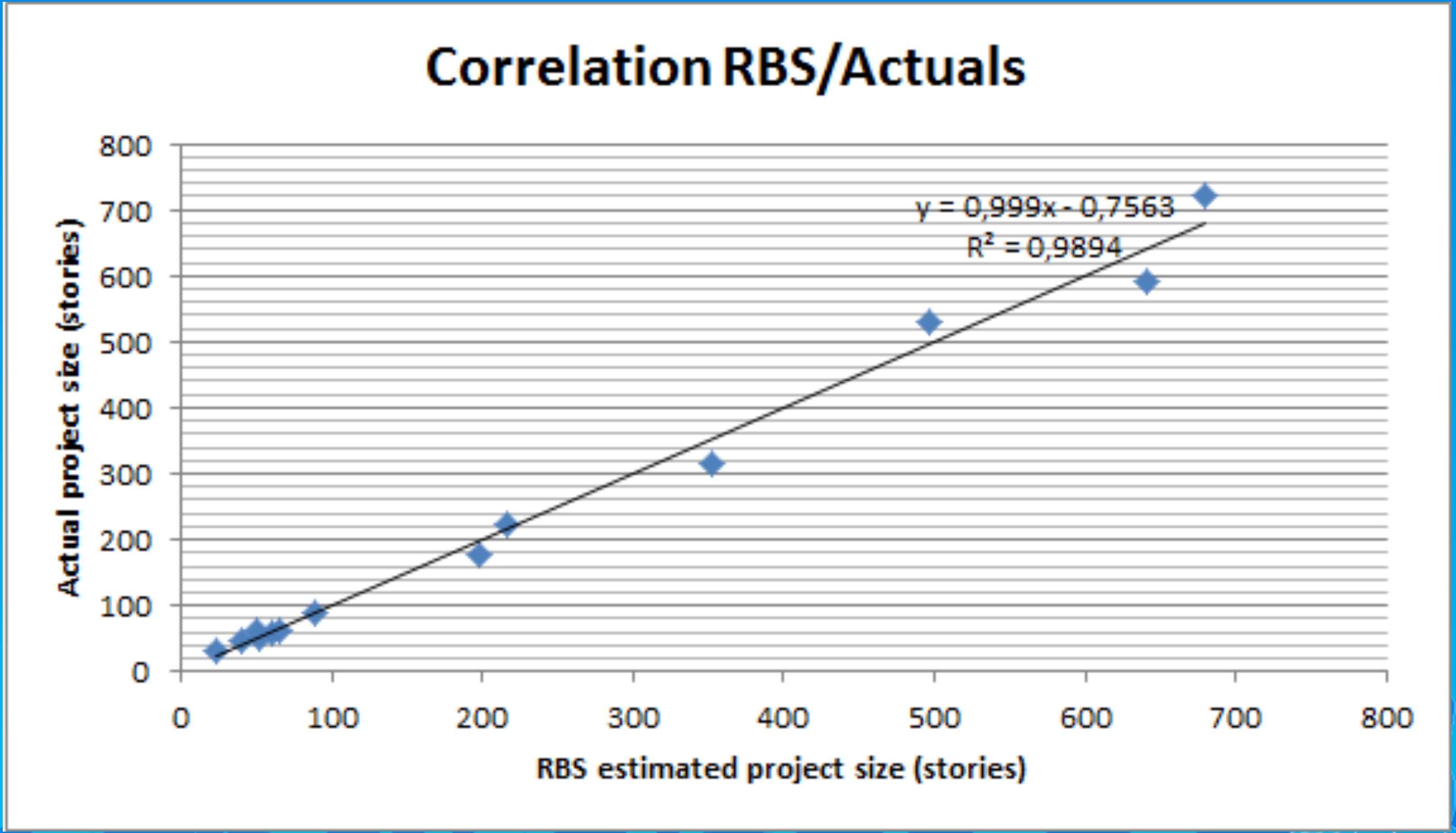
RBS estimates compared to the actual results of 13 real ScrumDo.com projects

ScrumDo

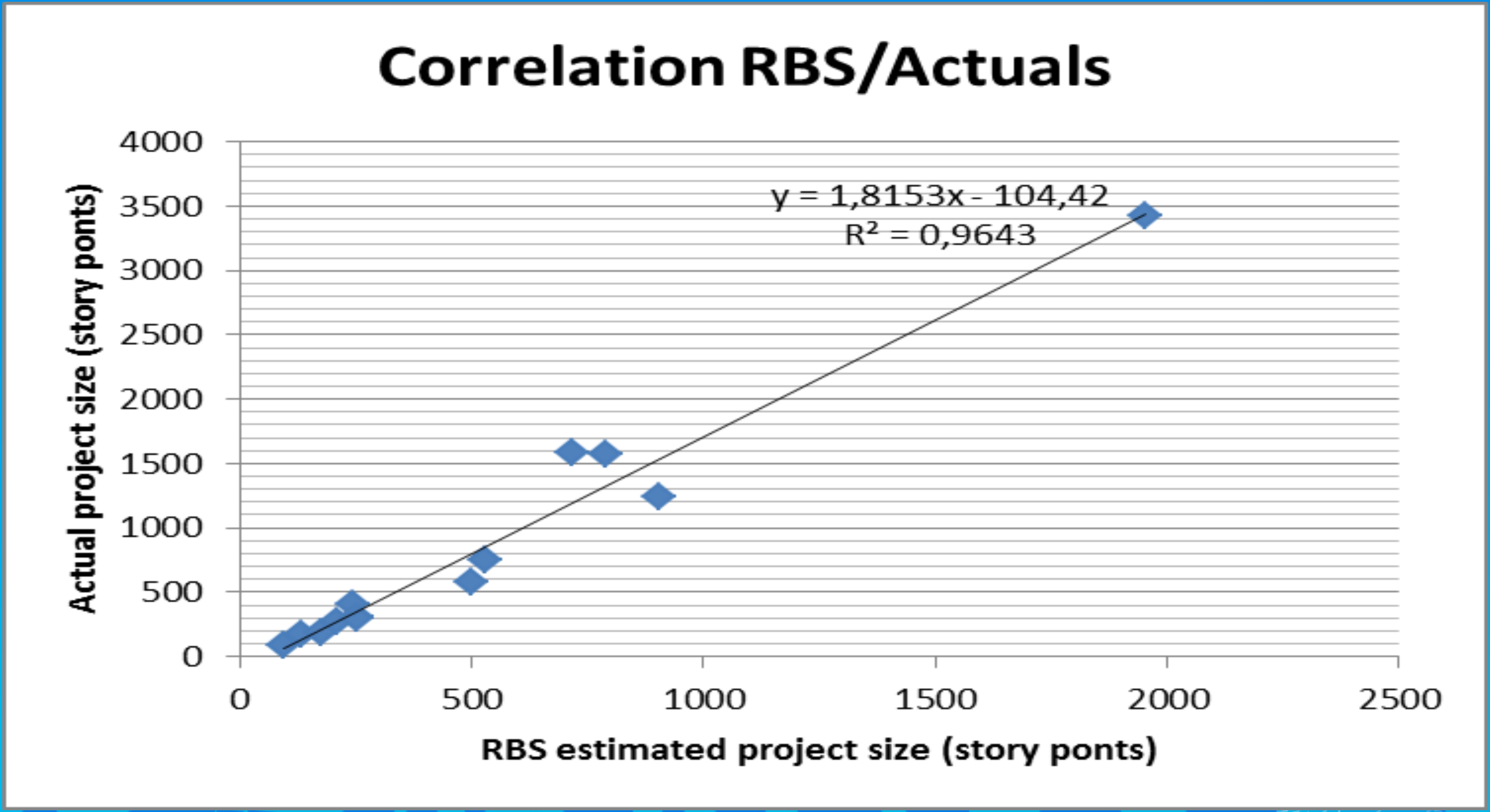
Common for all 13 projects

- Epic-Story-Task breakdowns
- Successful release history
- Stable teams (systems)
- Have an active ScrumDo coach or scrum master
- Commercial projects
- Have a minimum size of 12 epics/features.

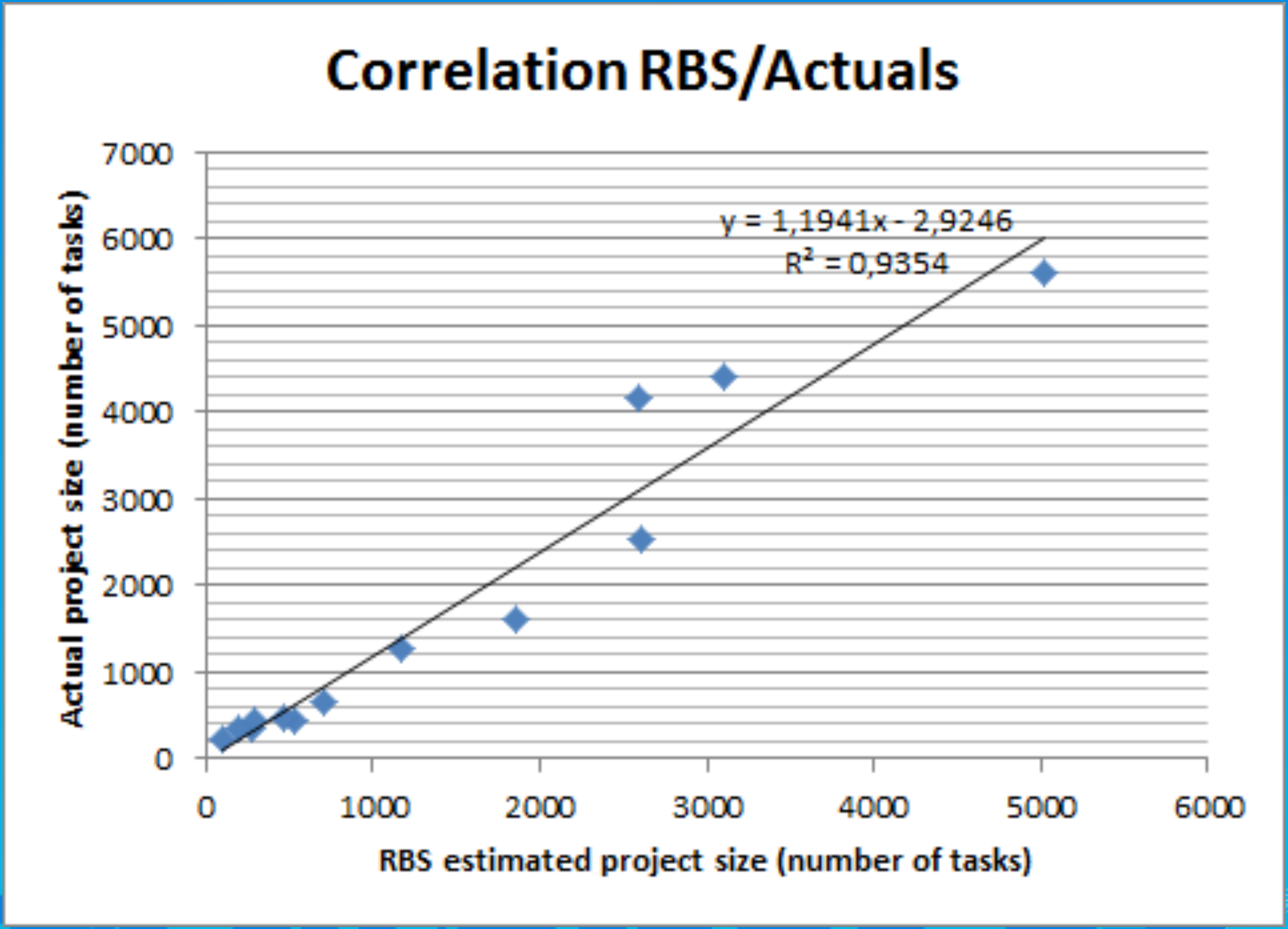
RBS estimated number of stories



RBS estimated total story points



RBS estimated number of tasks per story



Conclusions from Scrumdo.com data

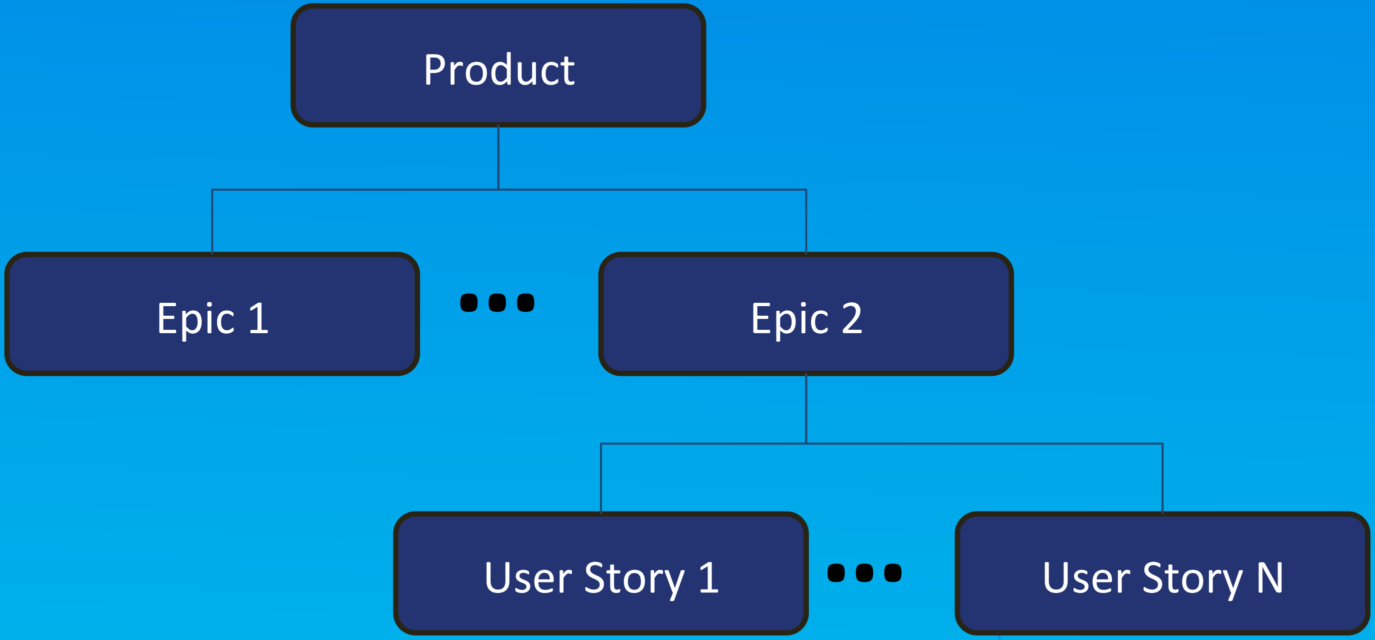
- During project execution all project teams consistently applied **a methodology** for slicing the requirements into user stories and sizing them using **story points & tasks**
- During project execution all project teams maturely managed the emergent and high-change-risk requirements
- Execution is more important than planning!!!

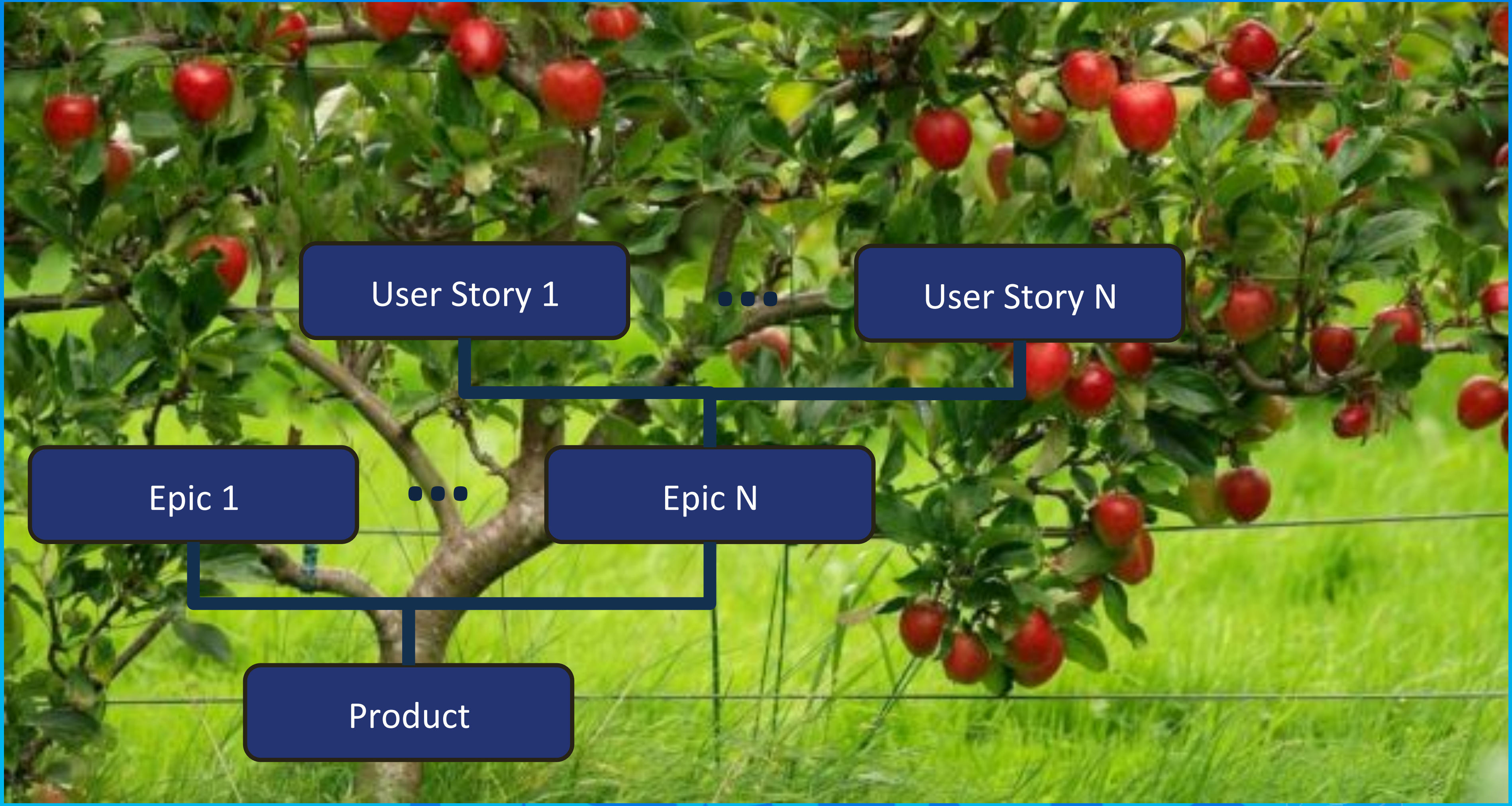
RBS for sizing new projects:

1. Applying RBS for estimating total number of user stories in a project
2. Applying RBS for estimating total Story points in a project
3. Applying RBS for estimating total number of tasks in a project
4. Applying RBS for estimating total number of BDD scenarios in a project

Applying RBS for estimating total number of user stories in a project

Stories based sizing model




User Story 1
...
User Story N
Epic 1
...
Epic N
Product

Mapping

Product	Trunk
Epic	Branch
User Story	Terminal Shoot

RBS estimate of the of total number of user stories for a project

$$\hat{X}_i = \frac{\begin{pmatrix} Number\ of\ user\ stories \\ in\ the\ sampled \\ Epic \end{pmatrix}}{\begin{pmatrix} \dfrac{1}{\begin{matrix} Number\ of \\ Epics \\ in\ the\ project \end{matrix}} \end{pmatrix}} \qquad (1)$$

Where:

$\hat{X}_i$ is an estimate of the total number of user stories for the project.

Total number of user stories for the project

$$\hat{X} = \frac{1}{m}\sum_{i=1}^{m}\hat{X}_i = \frac{1}{m}\sum_{i=1}^{m}\frac{\hat{S}_i}{\frac{1}{n}} \quad (2)$$

$\hat{X}$ is an unbiased estimator of the total number of user stories for the project

$\hat{S}_i$ is the number of user stories in the m-th epic

m is the number of estimates done

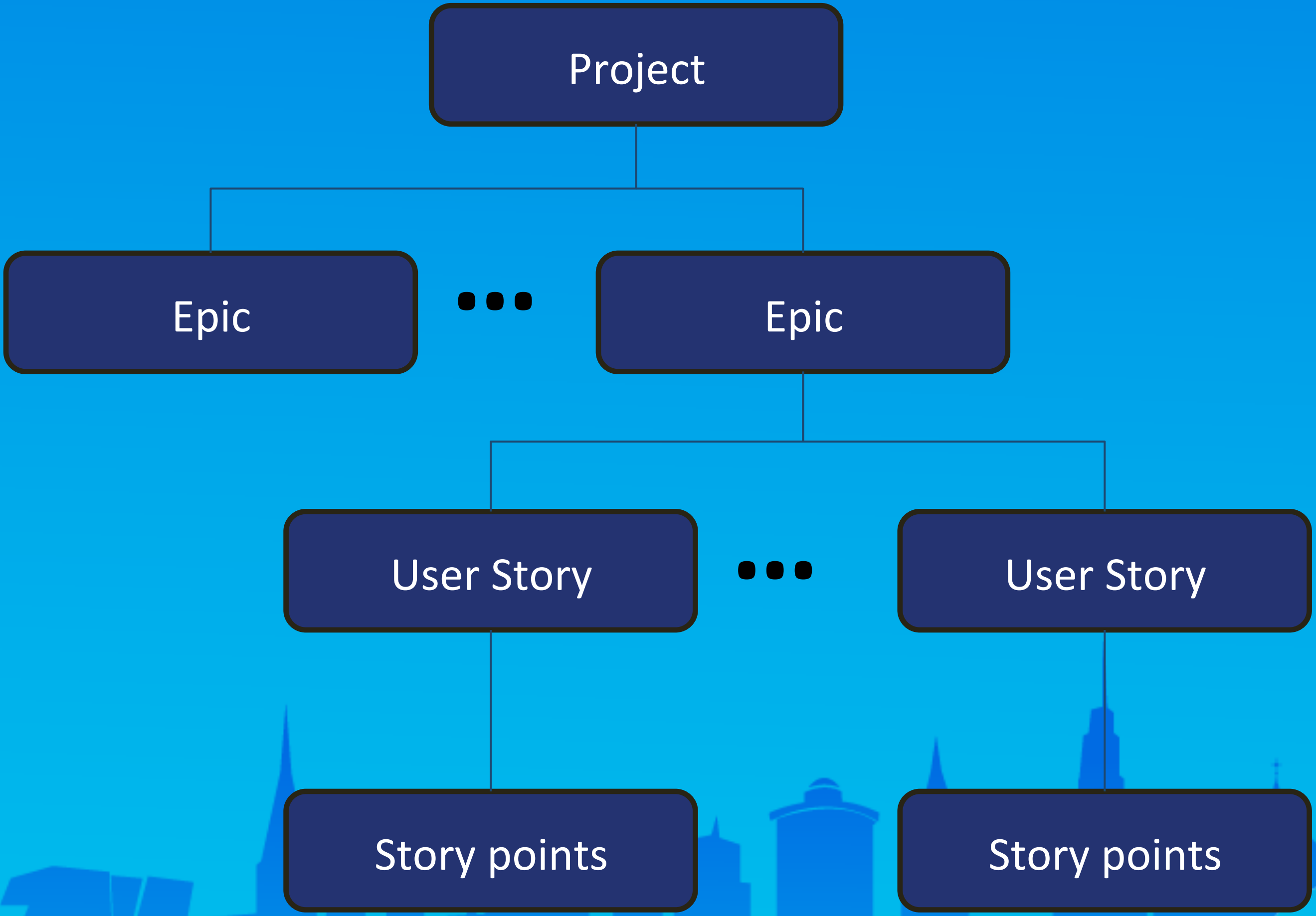
n is the number of epics in the project

Algorithm

1. Divide the project scope into epics.
2. Randomly sample one of the epics
3. Analyze how many stories are in the sampled epic. Write down the number of stories.
4. Using formula (1) calculate one estimate of the total number of stories for the project
5. Repeat points 2-4 between 7 and 11 times
6. Using formula (2) calculate the total number of stories for the project

Applying RBS for estimating total Story points in a project

Story points based sizing model


Project

Epic

...

Epic

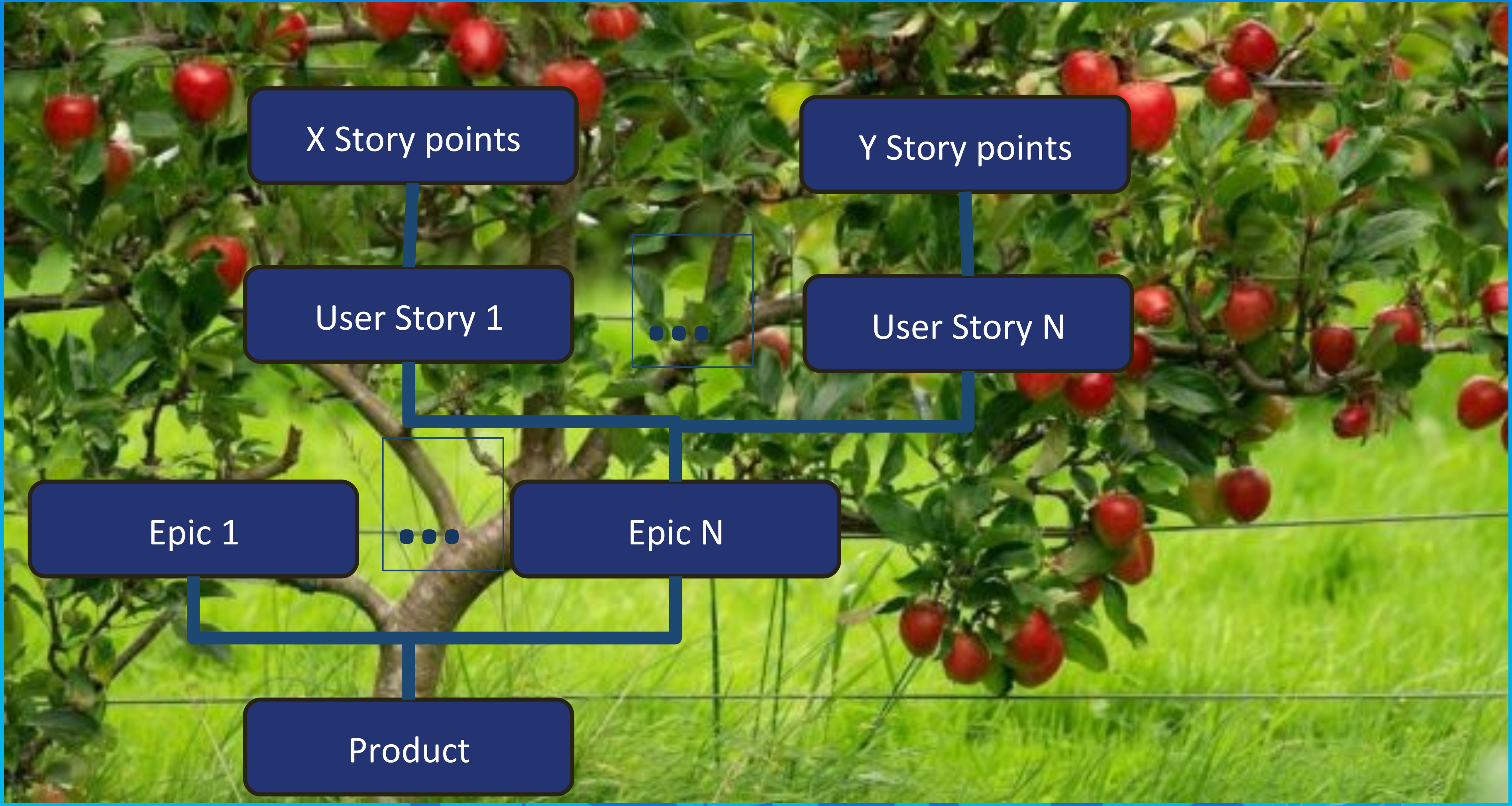
User Story

...

User Story

Story points

Story points



X Story points

Y Story points

User Story 1

...

User Story N

Epic 1

...

Epic N

Product

Mapping

Product	Trunk
Epic	Branch
User Story	Terminal Shoot
Story points per story	Number of Fruit on the Shoot

Estimate of the of total story points for a project

$$\hat{X}_i = \frac{\begin{pmatrix} Story\ points \\ of\ the\ sampled \\ User\ story \end{pmatrix}}{\begin{pmatrix} \frac{1}{\begin{matrix} Number\ of \\ Epics \\ in\ the\ project \end{matrix}} \end{pmatrix} \begin{pmatrix} \frac{1}{\begin{matrix} Number\ of \\ User\ stories \\ in\ the\ sampled\ Epic \end{matrix}} \end{pmatrix}} \quad (3)$$

Where:

$\hat{X}_i$ is an unbiased estimator of the population total of the of story points for the project.

Total story points for the project

$$\hat{X} = \frac{1}{m}\sum_{i=1}^{m} \hat{X}_i \quad (4)$$

Where:

$\hat{X}$ is an unbiased estimator of the total story points for the project.

m is the number of estimates done

Algorithm

1. Divide the project scope into epics.
2. Randomly sample one of the epics
3. Analyze how many stories are in the epic. Write down the number of stories.
4. Randomly sample one of the stories of the epic from p.2
5. Estimate the story points for the story from p.4
6. Using formula (3) calculate one estimate of the total story points for the project
7. Repeat points 2-6 between 7 and 11 times
8. Using formula (4) calculate the total story points for the project

CONCLUSIONS

- RBS is a forecasting technique for sizing software projects without prior identification, analysis and sizing of every single user story. Project size may be measured in story points, number of tasks, BDD scenarios.
- By running RBS on past data from actual projects, we found that the RBS would have estimated the same size without all the usual effort.
- RBS helps us to reduce uncertainty regarding "how much" software needs to be developed when we have to make portfolio related decisions, provide quotations on prospect projects etc.

More on the topic…

Probabilistic Project Sizing Using Randomized Branch Sampling (RBS)

@dimiterbak

Dimitar Bakardzhiev is the Managing Director of Taller Technologies Bulgaria and an expert in driving successful and cost-effective technology development. As a LKU Accredited Kanban Trainer (AKT) and Brickell Key Award 2015 Finalist, Dimitar puts Lean principles to work every day when managing complex software projects. Dimitar has been one of the evangelists of Kanban in Bulgaria and has published David Anderson's Kanban book as well as books by Goldratt and Deming in the local language.

goto; conference

Please

Remember to rate this session

Thank you!

Join the conversation #gotocph