November 2023

Forecasting Engineering Team Capacity and Work Estimates Using Product Roadmaps

Saif Khalaf

Mailys Levassort

Amisha Singh

Kimberly Reardon

Forecasting Engineering Team Capacity and Work Estimates Using Product Roadmaps

ABSTRACT

Despite substantial research, deriving work estimates and team capacity for a given software development project continues to be notoriously difficult to get right. This disclosure describes techniques to automatically and rapidly calculate, in the same unit of measure, engineering team capacity and engineering work estimates for a given set of product features. With use of the described techniques, software development projects can be better organized, and engineering talent can spend time on engineering problems rather than on planning. The automated team capacity and work estimate calculations can be more accurate than typical human-generated estimates, and lead to more accurate product development decisions. Planning data across the development organization is standardized into a common data model to support deeper analysis. Historic team capacity and work estimate data is leveraged to accurately and automatically predict future work and resource estimates using heuristic and statistical techniques. Automated, tailored communication including structured planning data, actions, and insights is delivered to responsible individuals or groups in a timely manner.

KEYWORDS

- Project management
- Software development
- Agile development
- Scrum software development
- Burndown chart
- Operations research
- Software development velocity

BACKGROUND

When planning the development of a software project, two values are compared over time - the capacity of an engineering team projected through the duration of the project and an estimate of the engineering work over the same duration. Both team capacity and work estimates are in the same unit of measure.

Despite substantial research over the last few decades, deriving work estimates and team capacity for a given project continues to be time consuming and notoriously difficult to get right, e.g., to achieve meaningfully accurate predictions of capacity and work estimates over the project duration.

Software development projects rely on accurate predictions of development times and staffing requirements to deliver on a strict schedule. On-time delivery of milestones is especially critical when such projects intersect other projects, e.g., hardware development for hardware that the software is to run on. To ensure on-time delivery, engineering managers need to devote a large proportion of their time to planning and estimating, which is a loss of valuable productivity on the part of the most experienced engineers of an organization. It is more optimal for engineers to solve engineering problems than to keep an account of their team members' availability or to predict engineering effort based on a product description or specification.

Agile and scrum software development methodologies were developed partially in response to the challenge of accurately predicting timelines for software projects. Some planning tools offer burndown charts that extrapolate a calculated team velocity, but rely on human-generated estimates of effort, which tend to be a major source of error. While retrospective views of progress are available, forward-looking projections are based on manually entered (and error-prone) estimates of staff loading.

DESCRIPTION

This disclosure describes techniques to automatically and rapidly calculate, in the same unit of measure, engineering team capacity and engineering work estimate for a given set of product features. Software development projects can be better organized, and the pressure on engineering talent to spend large portions of their time planning is relieved. The automated calculations for team capacity and work estimates are more accurate than human-generated estimates, and lead to better, more accurate product development decisions.

The techniques leverage certain observations, such as:

- Planning data across an entire organization can be standardized into a common data model to support deeper analysis.
- Historic team capacity can be an accurate prediction of future team capacity if the team size remains the same; otherwise, historic data can be adjusted proportionally to the changes of staff size.
- Historic work estimate data can be an accurate prediction of future work if future work can be accurately attributed to similar past work.

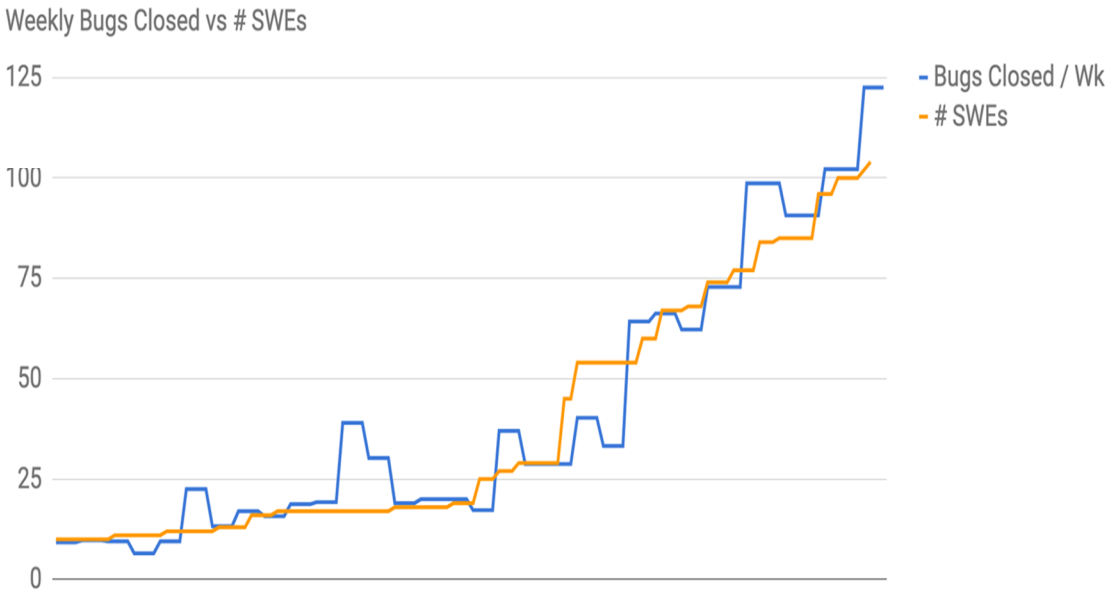

Fig. 1: Rate of issue closures closely tracks the number of engineering resources

- Engineering teams using a ticketing system naturally develop a predictable pattern of work completion rates, which can support future predictions. For example, Fig. 1 illustrates the number of issues closed per week (blue curve) against time. Also illustrated (orange curve) is the number of software engineering (SWE) resources against time. A close correlation is observed between the rate of issue closure and the available number of engineering resources.
- Mapping of past performance to potential future work to enable automated estimation can be done with heuristic and statistical techniques.
- Rule-based automated corrections to key planning metadata enable the estimation model to yield usable results.

- Automated, tailored communication that leverages structured planning data can deliver relevant actions and insights to individuals or groups in a timely manner

Historic ticketing system data is used to automatically calculate a prediction of team capacity for multiple interacting teams within an organization. Completed work, captured in a structured format, is used to provide a prediction of work estimates for product feature requests. Both team capacity and work estimates are recalculated periodically and scale to the changing requirements or behaviors of an organization. Team capacity and work estimates are initially calculated using heuristics, and once set up, are updated by leveraging statistical averaging and statistically significant quantities to yield usable predictions.

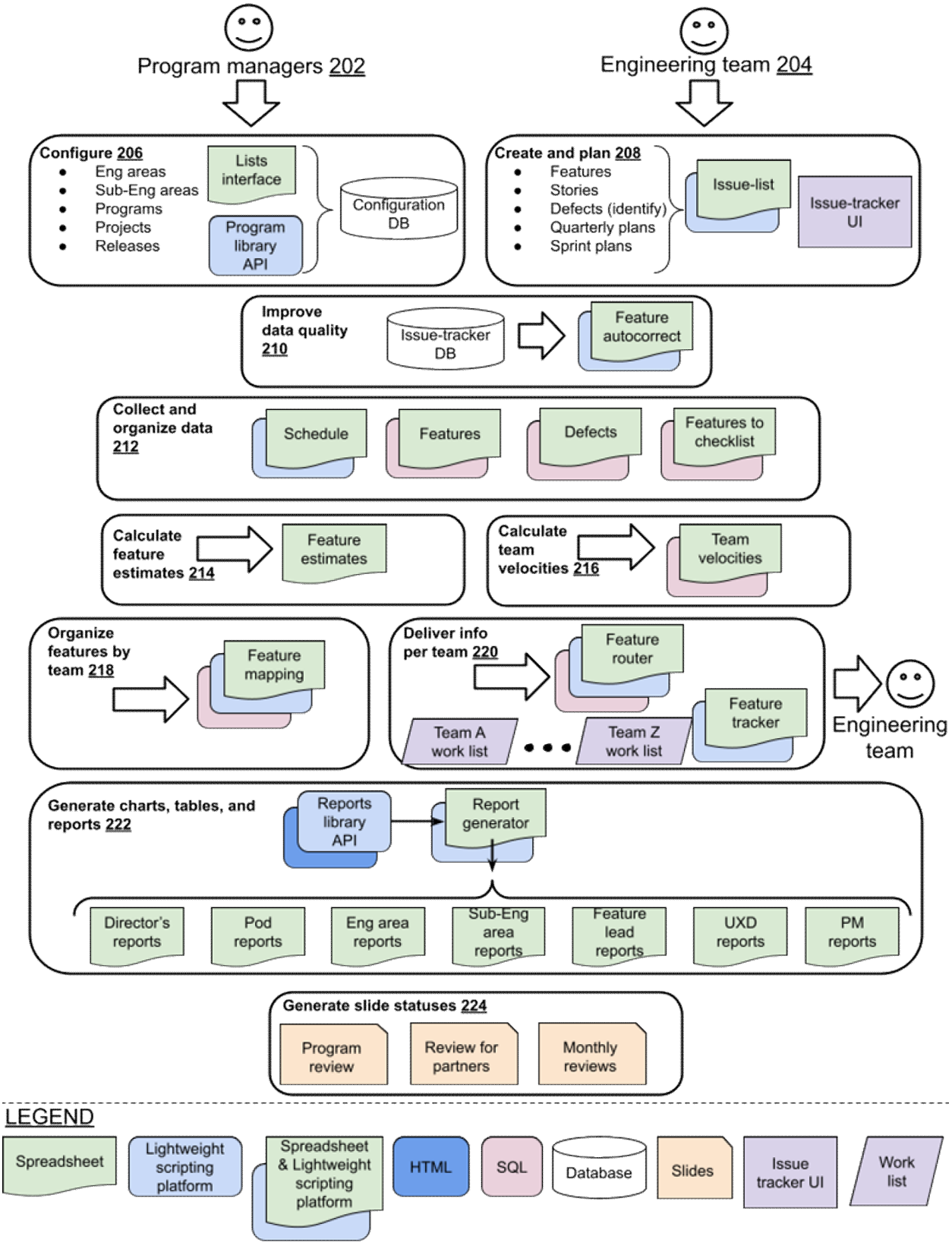

Fig. 1: Forecasting engineering team capacity and work estimates using product roadmaps

Fig. 1 illustrates forecasting team capacity and work estimates using product roadmaps. A team of technical program managers (202) configures the product roadmap into a configuration database (206). Configuration parameters can include engineering areas, sub-engineering areas, programs, projects, releases, etc., and tools used to configure the database can include a list interface (via, e.g., a spreadsheet program) and a program library API (e.g., via a lightweight scripting platform).

In parallel, an engineering team (204) plans out the product development (208). Planning the product development can include the identification of product features, stories, defects, quarterly plans, sprint plans, etc. Tools used to plan the product development can include an issue tracker, interfaced using an issue list presented via a spreadsheet and a lightweight scripting platform.

A history of issues, stored in an issue tracker database, is accessed to improve data quality by auto-correcting features (210). Data input by the program managers and by the engineering team is collected and organized into schedules, features, defects, checklists, etc. (212). Tools used to organize and collect data include spreadsheets, used in combination with lightweight scripting platforms and structured query language (SQL).

Prior statistics of issue closure rates as a function of engineering resources (e.g., as illustrated in Fig. 1) can be used to forecast feature estimates (214) and team velocities and forecasted future capacity (216). Example tools that can be used to forecast feature estimates and team velocities include spreadsheets and SQL.

Features are mapped and organized by team (218). A feature router can be used to deliver mapped features to each team in the form of work lists. Example tools that can be used to map features include spreadsheets, SQL, lightweight scripting platforms, etc. Example tools that can

be used to construct a feature router include spreadsheets, SQL, lightweight scripting platforms, etc.

With feature (work) estimate forecasts and team capacity available in the same units, charts, tables, and reports can be generated by a report generator (222). The report generator can rely on a reports library API (e.g., built using lightweight scripting platforms and HTML) to generate reports. The reports are tailored and distributed to specific audiences, e.g., director's reports, engineering area reports, sub-engineering area reports, feature lead reports, user experience designer reports, program manager reports, etc.

Similarly, slide sets are tailored and distributed to specific audiences (224), e.g., slides for program review, slides for external and internal partners, slides documenting monthly progress, etc.

CONCLUSION

This disclosure describes techniques to automatically and rapidly calculate, in the same unit of measure, engineering team capacity and engineering work estimates for a given set of product features. With use of the described techniques, software development projects can be better organized, and engineering talent can spend time on engineering problems rather than on planning. The automated team capacity and work estimate calculations can be more accurate than typical human-generated estimates, and lead to more accurate product development decisions. Planning data across the development organization is standardized into a common data model to support deeper analysis. Historic team capacity and work estimate data is leveraged to accurately and automatically predict future work and resource estimates using heuristic and statistical techniques. Automated, tailored communication including structured planning data, actions, and insights is delivered to responsible individuals or groups in a timely manner.

REFERENCES

1. “Jira | Issue & Project Tracking Software | Atlassian” available online at https://www.atlassian.com/software/jira
2. “Portfolio for Jira | Atlassian” https://www.atlassian.com/software/jira/portfolio/.I'm