

FOCUS: Technical Debt

Interview Guide

Ward Cunningham termed “technical debt” as a metaphor for representing the consequences of making the trade-off between designing/implementing a fully-architected, sustainable software system and taking shortcuts to deliver the system quickly to the customer. The purpose of this interview is to identify how projects characterize, decide, to incur and manage technical debt.

1. Context

1.1. Describe yourself:

- 1.1.1. How many years of experience do you have?
- 1.1.2. What is your role on the project?
- 1.1.3. What are your responsibilities?
- 1.1.4. How many years have you worked on the project?

1.2. Describe your project:

- 1.2.1. What does the system do?
- 1.2.2. How old is the system?
- 1.2.3. Who are your customers (market segment)?
- 1.2.4. What is the software development process on the project?
- 1.2.5. What is the release planning process on the project?

2. Technical Debt

2.1. Describe an example of technical debt on your project.

- 2.1.1. What happened?
- 2.1.2. Why do you consider it to be “technical debt”?
- 2.1.3. Did you plan to incur the “technical debt”?
 - 2.1.3.1. If yes...
 - 2.1.3.1.1. Who made the decision to incur the debt?
 - 2.1.3.1.2. When did you decide to incur the debt?
 - 2.1.3.1.3. Why did you decide to incur the debt?
 - 2.1.3.1.4. How did you decide to incur the debt?
 - 2.1.3.2. If no...
 - 2.1.3.2.1. Who identified that this is a

debt?

2.1.3.2.2. When did you discover the debt?

2.1.3.2.3. Why did you decide that is a debt?

2.1.3.2.4. How did you discover the debt?

2.1.4. What is the impact of the technical debt on your project...

2.1.4.1. For the customers?

2.1.4.2. For development?

2.1.4.3. For future modifications to the system?

2.1.5. Do you know how much of the system is affected by the debt? If so, how do you measure the size of the debt?

2.1.6. How are you managing the debt? Are there any plans to pay off the debt?

2.1.7. How do you communicate the debt to your customers? Your managers? Your developers?

2.1.8. What are the benefits of incurring the debt?

2.1.9. What are the consequences of incurring the debt?

2.1.10. Was incurring the debt worthwhile? Would you do it again?

2.1.11. What did you learn from this experience? What would you do the same/differently?

Are there any other thoughts, comments, suggestions, and/or lessons learned from your experiences with technical debt that you would like to share?

This material augments the article, “A Balancing Act: What Software Practitioners Have to Say about Technical Debt” (IEEE Software, November/December 2012, pp. 22–27).

Unless otherwise stated, bylined articles, as well as product and service descriptions, reflect the author’s or firm’s opinion. Inclusion does not necessarily constitute endorsement by the IEEE or the IEEE Computer Society.