Welcome to GraphITs.Tech!

Welcome to the strange and wondrous land of GraphITs.Tech!

In this place software engineering is generated by self-described engineering geniuses (although none of them ever agree with any other and are not at all timid about vocalizing their differences). Interns learn a lot of things, some of them are even correct. Developers rail at sloppy code, ignorant of the memory that they themselves originally wrote said hacks. Project managers are cyclicly anxious, despondent, reassured, and euphoric. Sometimes even salespeople verge on technical hysteria. All this is overseen by a bipolar management team. Some members of this team have commented that REST endpoints sound relaxing, others have been know to say "vixen" when they mean "maven", and one or two need to be reassured that a stack overflow isn't going to get their shoes wet. Talking to management, the engineers mentioned above will sometimes ask if "B2B clients" means "business to business clients" or "2859 clients" (after converting hexadecimal to decimal in their heads). In short, a development environment which is somewhat less dysfunctional than the norm.

In other ways, too, GraphITs is unlike other companies. It works mainly on projects for which the objects represented by the data are strongly connected. It offers a variety of services like fraud detection and recommendations, or the technical architecture of social networks. Though grumbling can sometimes be heard, and is sometimes impossible not to hear, everyone takes quiet pride in belonging to this innovative company.

And it is in this raucous climate, on a beautiful morning, in a beautiful month, in a beautiful year, that a new client crosses the doors of GraphITs. This client has a need, and that's good. Because here, they just love to meet needs.

The client in our story is a technical book publisher known as ACME Publishing. ACME Publishing wanted to meet the GraphITs team in order to explain problems it has been experiencing with their data. And this is where our story begins. Once upon a time...

Characters

Varsha, *salesperson*

Jackie, *project manager*

Tiago, Iryna, *technical experts*

Ashley, *DevOps and network expert*

Brian, *trainee*

Client, ACME Publishing, a company specializing in technical book publishing

Warning › *We have purposely exaggerated the personality of each character for educational, and, we hope, entertainment purposes. Any resemblance to actual persons, living or dead, or actual events is purely coincidental.*

1. Kickoff meeting

Varsha, salesperson at GraphITs.Tech, greets the client in the large meeting room. The entire team is present.

Varsha: Good morning and welcome to GraphITs Services. We're delighted to meet with you and eager to hear about your needs and concerns. Help yourself to the doughnuts and coffee.

ACME Publishing: Hi. Thanks for the warm welcome. As you may know, ACME Publishing is becoming the uncontested leader in technical book publishing in the domain of computer science. These books are published in several formats such as paper, EPUB, PDF and HTML. We are also beginning to translate our books for export to other countries. We'd like to make improvements to our information system and update it to meet new challenges.

Varsha: Good, and what was it that drew your interest to us?

ACME Publishing: We've heard you're working in NoSQL data stores?

Varsha: That's right. But is there something in particular that leads you to believe that NoSQL will help you solve your problems?

ACME Publishing: Actually I'm not too sure, but I'm even less sure what the future holds. In short, I've heard that NoSQL could be a foundation for a more agile, more flexible system. As it stands, every time we need to change the structure of our SQL databases, it hurts like the dickens!

Varsha: That's partly true. But it depends to some extent on what we are talking about. NoSQL does not remove the need to have a well thought-out upstream design. A NoSQL system can simplify certain tasks, such as structural changes. But again, the mileage you get may vary depending on the details of your data.

REMINDER: The four types of NoSQL

- *Key-Value store*, data stored as key-value pairs;
- *Column store*, data stored as tables (also known as *Big Table*) shaped as columns and super columns;
- *Document store*, data stored as compound objects (generally as JSON or XML);
- *Graph store*, data stored as graphs.

ACME Publishing: So you do deal with graph storage, don't you?

Varsha: Correct. But please continue.

ACME Publishing: Well... other companies seemed quite uncomfortable when we've talked about data structures that haven't been completely designed yet. They propose that we store some data in a relational database, and the rest of unstructured data in a NoSQL system.

Varsha: That is quite common, to take advantage of both types of databases in information systems.

ACME Publishing: Well, I'd prefer a transactional system. And the rigid aspect of relationships in SQL databases slow us down when building new services.

Varsha: You seem to be surprisingly aware of technical problems!

ACME Publishing: As I said before, we've had major problems making changes to our applications. My team is pretty skilled, and they've briefed me about problems they've encountered.

Varsha: Terrific! I suggest that you detail what you'd like to do and then we'll set up a round table presentation of our team.

ACME Publishing: Here, in a broad outline, is our requirement: in order to better manage our growing orders, I would like to implement a process that would allow our customers to track the status of their transactions, from order to shipping. Also I'd like to see my catalog of books related to the order process. There are fuller details in some documents we have brought with us, but that's the 10,000 foot view.

Varsha: Of course. I think we've captured the essence of it for now. Let me introduce the members of our team. This is Jackie, who will act as project manager. Jackie, any opinions you'd like to share at this point?

Jackie: Good morning. It seems to me that this isn't a single requirement but several smaller ones. If I understand correctly you need a catalog of books and a way to manage the order process.

ACME Publishing: Exactly.

Varsha: This is Tiago, Iryna and Ashley, our technical experts.

Tiago, Iryna and Ashley: Hi!

Varsha: And this is Brian, our wannabe developer... hey, where is Brian?

Ashley: In the kitchen, of course!

Varsha: Hmmm, Brian is one of a kind. OK, guys, how about you start working on the first drafts. Let us know when you have an implementation proposal.

ACME Publishing: Isn't my requirement a bit too "classical" for your team?

Varsha: We hope to prove to you that graph databases can not only solve a classical requirement, but then can also ease system maintenance!

Jackie: Yes, it's well known that Neo4j, being a NoSQL database, allows to start with data structures yet unrelated and imperfect, then evolve to more polished things. Neo4j support database schemas with strong constraints. Moreover, it is also a transactional ACID database, therefore highly coherent, which also offers a particularly advanced relationship management between entities. But whoever runs can also walk... usually...

ACME Publishing: If I understand well, the advantages of NoSQL without the downsides.

Jackie: Well, imagine that in the past your goal was only to store data. Now, your goal is to do more. Hence your presence here.

ACME Publishing: You sure are selling me a good night's sleep and sweet dreams. But can your technical teams deliver? How can I know my project will be successful?

Varsha: Expertise is our team's strong point, which is why you came to us. Once you've been shown the power of our technology solution, we think there's a good chance that you will decide to employ our graph database technologies. Our job will be to show you, step by step, how this success is possible. Once you've seen that by approaching the problem differently (and with fewer technical constraints), you'll be pleasantly surprised by the results.

ACME Publishing: Well, I'll let you get to work. Varsha, let's you and I talk contracts.

2. Debriefing

Varsha, Jackie, Tiago, Iryna, Ashley and Brian gather to discuss the project roadmap.

Varsha: So, what are your thoughts about this project?

Tiago: For the catalog part, not very complicated. We could almost do that with a relational database.

Iryna: Why not go with CSV files, while we're at it?! Even if that's true, we're not going to do it like that. Who wants to manage two systems when one can do the job?

Jackie: I'm with Iryna. Just because it could work with a SQL database doesn't mean we have to go that way, even if the advantages of storing the catalog in a graph is less obvious than for the other scenarios. Why don't we talk about the technology stack?

Tiago: As for the catalog, the domain model isn't going to be very dynamic, so we could use OGM (Object Graph Mapper) or even SDN (Spring Data Neo4j).

Jackie: Cool. What about the rest?

Iryna: For the other parts, I suggest we use CYPHER directly, without necessarily using *object mapping*; potentially we'll use APOC procedures[1], to optimize processing between Neo4j and our business services. We could use Bolt as the database driver, which is more efficient than the HTTP API.

[1] Awesome Procedures On Cypher

Jackie: But what if they want to connect it to their BI tools?

Iryna: CYPHER requests over JDBC should be fine, but we'll double check that.

Jackie: They will surely have legacy data to ingest...

Tiago: We have several options in Neo4j to do some data import/export, even on large volumes.

Ashley: I hate to bring this up guys, but the client wants to go international with several languages!

Tiago: So what's the problem?

Ashley: Have you heard of Europe? This minor continent with a 6 hours time difference from here.

Jackie: I still don't get your point.

Ashley: My point is that it's always the high-traffic time somewhere on the globe. We really can't afford to be down any time of the day. Which means High Availability (HA) which means clusters of Neo4j servers and hot backups.

Jackie: Not a bad point!

Brian: Wow, there's a sh...buttload of stuff to do. Where do we even begin?

Tiago and Iryna: With some methodology and design!

Iryna: This is going to be a blast. Once in a while we might even be on the same page.

Varsha: OK, I see you are off and running. I'll come back later for the next step.

The agenda: next steps

- Some methodology and design: Or how to go from requirements to graph structure design;
- Interactions with Neo4j: About setting up systems to talk and interact with Neo4j;
- Data import: For each graph, import legacy data;
- Operations: How to achieve high availability with clusters of Neo4j servers;
- Security: Securing the cluster and its data.