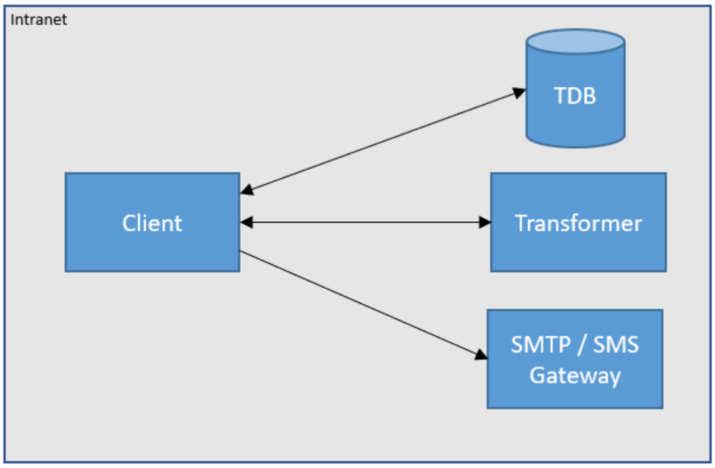
Welcome

to this system architecture migration task.

Please remember: the result is only the starting point for an upcoming discussion – there is no "correct" answer. However, you should be able to give a reason for your decisions.

Baseline

We have an existing output management system. This system enables a user of a client software to create styled templates (some kind of XML), store and load these templates in a central database, use loaded templates to create mail objects, PDFs and SMS texts (while providing variables) and send the result by SMTP or some public 3rd party SMS service. The image shows a simplified overview. PDFs can be sent attached to a (previously generated) mail.



The **client** is an Eclipse-RCP application on a Windows fat client. This client can be used to create mail, PDF and SMS templates. Internally, it defines an XML including all styling information and a list of expected variables. The client can store the resulting templates in a central database called **TDB** by the colleagues. The templates can also be loaded from the database by a unique name. Templates can be sent to the transformer to create a mail, PDF or SMS. This information can be sent by SMTP or a public 3rd party SMS gateway. PDFs can be attached to mails.

The templates stored in the **TDB** consist of some XML-like format as well as meta data like name, id, timestamps, etc. Currently, around 12000 templates exist – most of them have not been used for years. 96% of all creation tasks use about 2% of the existing templates.

Loaded templates can be sent together with the required variables to a **transformer** webservice. This service will interpret the style information and create a mail object, PDF (Base64) or plain text (for SMS).

Results are sent by **SMTP** or by some **SMS Gateway**.

New Requirements

This system worked for some time now, however, it always took manual effort to create and send mails. In the future, systems should be able to trigger document creation and send the results without human interaction.

For this, we need webservices offering the following functionality:

1) Create and send SMS
2) Create PDF
3) Create and send Mail (with or without attachment)

Creation needs a template name (to load the respective template) as well as variables (Key/Value).

Unfortunately, all backend systems (TDB, Transformer, SMS Gateway, SMTP server) are error-prone. However, it is required that a triggered execution with valid information will always result in a created and sent object.

Your Ideas

What do you suggest? What would the new architecture look like? Where do you see the need for clarification? What are the reasons for your decisions? What technologies would you recommend?

Please be as specific as possible. Your result might have any format. You may draw (photo) or write or whatever feels best for you.