Case Messaging: the message thread as a platform for dental services

A **Case Message** is a group message thread between stakeholders in a dental case. It uses a HIPAA compliant messaging system and a vendor-specific chatbot, always in the message, to deliver services to the stakeholders. A customer doesn’t order - they send a message. In this way messaging becomes a platform for collaboration between providers, vendors, and patients.

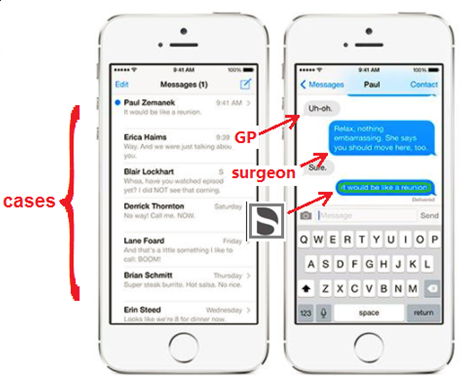


Figure 1 A messaging platform is ideal for archiving cases (on the left) and for reviewing or updating single cases (on the right). The vendor’s chatbot provides services inside the message

**Types of Services:** Ordering, order status, design services, planning services, image exchange, case notes, procedure FAQ, patient education, appointment reminders, co-scheduling, and other practice management functions.

**Types of Stakeholder**: Vendor, Lab, GP, Surgeon, Patient.

(A member of the public might access a non HIPAA compliant service – to ask questions or find a DS supplied GP. Also an insurer might want to “be in the conversation”. But that’s another story.)

**Value for the vendor**: The value of the message thread is at least the value of any order it contains. Also it is a premium service and supports higher pricing for the vendor’s parts. Still further, the ability to archive a case message as a single record provides traceability.

Auxilliary benefits include: increased brand awareness, opportunities to up-sell other services, and the value of vendor employees *themselves* using the messaging service with customers (J. Bergstresser said: “I could use that right now”). Finally, providing message based dental services is an opportunity to challenge the monopolies of practice management sofware (Shein and Patterson).

**Value for the provider or lab**: The value of having a vendor-specific chatbot handling the case is the huge convenience of having all the case information tracked within the thread and available at a word or at the flick of a finger.

**Chatbot Requriements:** The chatbot handles: Questions, Requests, Interactive Guidance. It will keep track of the particular treatment involved and of the current status of this message, will “listen” and respond in context.

**Notes:**

The message tracking number

When a message is initiated, it is assigned an internal tracking number (in the vendor’s system) and an external subject (could be a patient name) for use by the lab or provider. The tracking number coordinates the message over multiple vendor systems. For example, the identity of any part ordered within the conversaton will be tracked and accessible by the vendor using this tracking number.

Use Scenarios

A message between *any* of the types of stakeholders suggest possible services that could be useful to them. For example:

A member of the public is curious about implants and starts a non HIPAA compliant message with DS. They find a nearby GP who uses DS parts.

A GP has a patient needing an implant treatment and starts a case message with the Surgeon. Part ordering occurs naturally within the message. Schedules are coordinated. Order status is available on the stakeholders’ phones. Later, the patient gets an appointment reminder about pre-medication and the forms needed for the appointment (perhaps avoiding a costly no-show). All the materials, parts, *and information,* are delivered to the stakeholders as needed.

The next day, the GP reviews the status of various cases. With a smartphone in his pocket, he can skip the hassle of the office software.

A Lab needs to know the implant type. Since they are already in the message, they can ask the chatbot directly and, if it does not know the answer, it can forward the question.

A Lab has ordered an abutment designed by DS and wants to review order status, and then designs. A message thread keeps track of all exchanges, allows word-based 3D editing of the designs and other services.

Competitors

* Non HIPAA compliant systems.
* Strauman just released a GP-Surgeon collaboration tool.
* More general competition: practice management sofware, eg Dentrix (Schein) or Eaglesoft (Patterson), which are near monopolies.

**APPENDIX I– Messaging/Chatbot Architecture Dependencies**

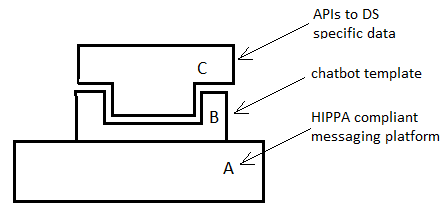
****

Figure 2 A chatbot on a messaging platform consists of platform (A) and a generic chatbot (B) connected to vendor-specific information (C)

1. A HIPAA compliant Message System

**Features**

* hosts Python textIn/Out chatbots
* supports voice-to-text
* services include: hosting, thread archive, logo display

**Costs**

* platform **license fee**
* voice-to-text **license fee** (may be included)

**Replacement costs**

* multiple providers exist, DS could build its own messaging system
* new providers should support Python chatbots
* replacing Python is more costly

1. Open source chatbot templates

**Features**

* Python
* textIn/Out
* vendor API “stubs”

**Costs**

* open source chatbot is **free** (first one anyway)
* upgrades and maintenance have **developer costs** (include in platform licensing fee?)

**Replacement costs**

* multiple providers exist, DS could write its own chatbots
* costs for new chatbot development
* some natural language understanding (NLU) requires license (Microsoft, Google)

1. Vendor specific API calls

**Features**

* standard APIs to implement chatbot template “stubs”

**Costs**

* minimal, if APIs exist

**Replacement costs**

* N/A

**APPENDIX II– LUI versus GUI** (Language UI versus Graphic UI)

A language interface, in the form of a chatbot inside a message, uses no screen real estate (a “single dot” UI) but it can act as a gateway to anything else. It can display links and can serve as a general proxy for traditional menus. The proposition is: anything you can do with a webpage you can do inside a message.

Language interfaces are:

* a pull technology; where a user asks for an option
* hands free (if voice-to-text is used)
* eyes free
* with a screen divided between data display and the message. No menus are needed.
* The message is its own record and the archive is readily available “one level up”.

Traditional webpages and Graphic Interfaces are:

* a push technology; where a user chooses from fixed options
* using hands
* eyes are required to find information on a complex screen
* with a screen divided between data display and menus. There is no record of the current task and no available archive of previous tasks. Email exchanges and phone calls may be informal.

The value of messaging with chatbots seems clear. In general, adding functionality within messaging makes sense anywhere that communication between people is part of the consumption of goods.

It is worth emphasizing that *the case message is a unique record of the case***.** It provides traceability and, for example, could be saved as a blockchain. By contrast, online orders, phone calls, and email exchanges do not form a single record. It is difficult and perhaps impossible to have a concept of a “case” unless there is some entity, some single record, to represent it. And since the concept of “case” is important to customers, how can a vendor conceptualize their needs or design solutions without some way to represent it? Further, it seems likely the case record will eventually be required, at least for liability purposes.

Complexity

Conventional vendor UI is subdivided by product types and immersed in the tabular forms of webpage and databases. Considering the different services that might be offered around those products allows us to slice the problem differently and, I would argue, more modularly and scalably than with a webpage. Roughly speaking, the “slicing” is driven by function rather than by content.

Much of what occurs on a webpage is simply reproducing what you would find anywhere online and which you can also find on a smartphone: acccount id, archive, review, create, etc. So why write that software that you could leverage from the smartphone? It should come as an immense relief to be able to offload large parts of the UI to a 3rd party.

That being said, the need to develop messaging services is an opportunity for software development teams to move beyond webpages and standard content packaging into areas of where the focus is on specific dental services. At the same time, other developers could focus on a corporate API backbone, to supply what is needed to the services.

Acceptance of Chatbots

Ultimately the acceptance of chatbots relies on the idea that a chatbot is more than an amusing parlour trick and that it can have a real, if primitive, understanding of language. This comes down to people wondering if a chatbot is even possible. To which one replies: of course it is. At least a chatbot can use structured commands. Done properly, a chatbot can immitate natural interactions and do useful work.