

UNIVERSIDADE DO MINHO
MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA

ENGENHARIA DE CONHECIMENTO

(2º SEMESTRE / 4º ANO)

Interoperabilidade Semântica - Benchmark de Envio de Mensagens HL7

Grupo 11

Alexandru Domente - PG41036

Francisco Reynolds - A82982

José Costa - A82136

Conteúdo

1	Introdução	2
1.1	Contextualização	2
1.2	Objetivos e Trabalhos Propostos	2
2	Arquitetura	3
2.1	Bases de Dados	3
2.2	Sistema A	4
2.3	Sistema B	4
3	Estudo de Complexidade	5
3.1	N Aberturas de Sockets TCP e Envio Sequencial	6
3.2	N Aberturas de Sockets TCP e Envio Batch	7
3.3	1 Abertura de Sockets TCP e Envio Sequencial	8
3.4	1 Abertura de Sockets TCP e Envio Batch	9
3.5	Compilação dos Resultados Obtidos	9
3.6	Análise dos Resultados Obtidos	10
3.7	Possíveis Melhorias	11
4	Conclusões	12
	Referências	13

1 Introdução

1.1 Contextualização

O presente relatório foi elaborado no âmbito da elaboração da ficha nº 2 da Unidade Curricular de Interoperabilidade Semântica, que se insere no 2º semestre do 4º ano do Mestrado Integrado em Engenharia Informática.

Após um pequeno *brainstorm* e debate do problema por parte de todos os elementos do grupo relativamente ao enunciado colocado pela equipa docente, foi exposto que a respetiva ficha tem, como propósito fundamental, utilizar o sistema já desenvolvido na ficha anterior desta Unidade Curricular e, a partir dela, desenvolver um pequeno sistema de geração de mensagens capaz de as enviar desde o Sistema A para o B, de modo a ser criado um *benchmark* que torne visível e mesurável a complexidade associada a este processo.

Como tal, o trabalho baseia-se inicialmente na geração de dados que irão constituir as mensagens a enviar, da mesma forma que serão também alocadas no Sistema A. De seguida, é exposto o procedimento da inserção da mensagem no respetivo sistema, bem como o procedimento de envio da mesma para o Sistema B. Após estes passos decorrerem com sucesso, é então ilustrada a confirmação de sucesso de receção e inserção da mesma no sistema.

Por fim, é feita uma análise exaustiva sobre a quantidade de dados enviados e o respetivo tempo de envio. Com esta informação, serão elaboradas tabelas que nos permitam perceber a complexidade do problema e a disponibilidade de dados com os quais o sistema por nós desenvolvido é capaz de lidar.

1.2 Objetivos e Trabalhos Propostos

Com este trabalho pretende-se, de uma forma concreta, obter informação estatística relativa à criação, inserção e envio de mensagens HL7 pelo sistema montado pelo grupo na ficha prática precedente. Com essas informações, o grupo será capaz de perceber onde residem os pontos fracos do sistema que limitam o fluxo de mensagens e ponderar quais os procedimentos adequados para os solucionar.

No global, o grupo tenciona colocar em prática procedimentos e conceitos lecionados ao longo das primeiras aulas de Inoperabilidade Semântica, bem como toda a bibliografia encontrada na internet sobre o problema, dado que o desenvolvimento desta ficha decorre, de forma excecional, remotamente.

No entanto, note-se que os conhecimentos adquiridos pelo grupo na área de desenvolvimento *web* serão uma componente bastante auxiliadora para o desenvolvimento desta ficha, visto que nos ajudarão a lidar com a construção do gerador de mensagens para a nossa aplicação, bem como todos os conceitos de bases de dados expostos até agora no perfil de especialização no qual nos encontramos inscritos.

2 Arquitetura

Abordando agora a arquitetura utilizada para a realização do trabalho, tal como indicado pela equipa docente, utilizamos 2 sistemas: A e B. Em cada um dos sistemas é utilizado um programa em *Python*, no caso do sistema A *sender.py* e no sistema B *receiver.py* e também uma Base de Dados MySQL em ambos os sistemas.

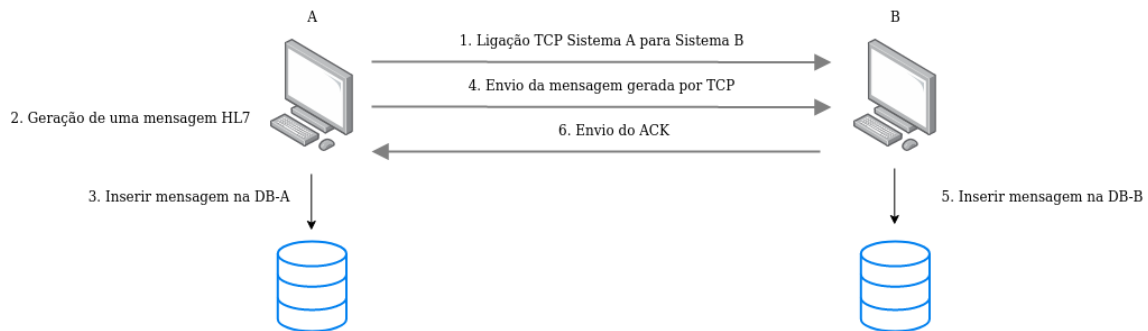


Figura 1: Arquitetura utilizada

2.1 Bases de Dados

Tanto a Base de Dados A como a B são iguais, têm apenas uma tabela chamada *worklist*, onde serão inseridos os dados gerados no Sistema A.

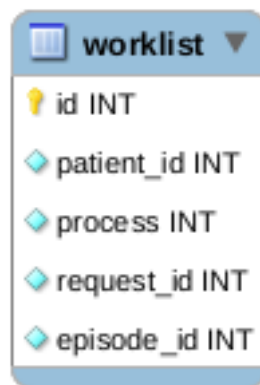


Figura 2: Esquema utilizado nas Bases de Dados A e B

Na tabela apenas consta o n^o de registo do pedido, o id do paciente, o n^o do processo, o id do pedido e o id do episódio. O grupo considerou que não é problemática a simplicidade da Base de Dados com a qual desenvolvemos este projeto, como o objetivo deste trabalho se foca na complexidade associada à geração e ao envio de mensagens HL7, e não na Base de Dados onde a informação que transporta será inserida.

2.2 Sistema A

O Sistema A tem as seguintes funcionalidades, asseguradas pelo *sender.py*:

1. geração de dados aleatórios para preencher os campos da mensagem HL7
2. inserção dos dados gerados na Base de Dados A
3. criação da mensagem HL7 implementando os campos previamente gerados
4. envio da mensagem HL7 e confirmação da inserção bem sucedida dos dados na Base de Dados B

2.3 Sistema B

O Sistema B tem as seguintes funcionalidades, asseguradas pelo *receiver.py*:

1. receção de mensagens HL7, recolhendo os dados que lá constavam
2. inserção dos dados na Base de Dados B
3. envio de uma mensagem informando o sistema B do sucesso ou do insucesso da inserção

3 Estudo de Complexidade

Para melhor estudar a complexidade à volta deste problema, investigamo-lo e chegamos à conclusão que para realizar o envio destas mensagens existem passos essenciais, nomeadamente:

1. geração dos dados + inserção na BD-A
2. criação da mensagem HL7 com os dados gerados
3. estabelecimento de uma conexão TCP entre os sistemas
4. processamento da mensagem HL7 no Sistema B
5. fecho da conexão TCP

Sabendo os passos que têm de ser realizados, criámos algoritmos que utilizando os passos acima mencionados, possibilitam o envio de mensagens HL7 e testam o sucesso ou insucesso das medidas implementadas que visam a comunicação mais eficiente das mesmas, através de estatísticas.

Dos passos acima mencionados, existem algumas otimizações que são fáceis de constatar como por exemplo:

1. dependendo do contexto, poderíamos realizar apenas uma abertura do Socket TCP e utilizar esse canal de transmissão continuamente.
2. o envio das mensagens poderia ser efetuado em batches e não sequencialmente

Formamos então os seguintes cenários de teste para saber como os fatores acima, influenciam o envio das mensagens HL7:

1. N Aberturas de Sockets TCP e Envio Sequencial
2. N Aberturas de Sockets TCP e Envio Batch
3. 1 Abertura de Socket TCP e Envio Sequencial
4. 1 Abertura de Socket TCP e Envio Batch

Seguimos também sempre a mesma abordagem em cada teste. Para cada cenário, efetuámos 5 testes, medindo o tempo que cada uma das tarefas acima mencionadas demoram. Chegamos também aos valores médios do tempo de execução de cada tarefa e cada teste, realizando também uma média do tempo que cada teste demorou, em função de quantas mensagens foram enviadas.

3.1 N Aberturas de Sockets TCP e Envio Sequencial

	Teste #1	Teste #2	Teste #3	Teste #4	Teste #5	Média	Média Percentual
Abertura de Sockets (ms)	0	0	0	0	0	0	0%
Fecho de Sockets (ms)	0	0	0	0	0	0	0%
Criação de Mensagens (ms)	11426	11411	11550	11427	13311	11825	19.83%
Geração e Inserção de Dados DB-A (ms)	21657	21977	22831	23296	22111	22375	37.52%
Processamento de mensagem no Sistema B (ms)	26506	26207	25272	24924	24287	25439	42.65%
Tempo de execução total (ms)	59589	59595	59653	59647	59709	59639	100%
Número de mensagens enviadas	1317	1311	1319	1295	1080	1265	-
Tempo médio por mensagem enviada (ms)	45.25	45.46	45.23	46.06	55.29	47.49	-

Tabela 1: Envio sequencial com N aberturas TCP

3.2 N Aberturas de Sockets TCP e Envio Batch

	Teste #1	Teste #2	Teste #3	Teste #4	Teste #5	Média	Média Percen- tual
Abertura de Sockets (ms)	0	0	0	0	1	0	0%
Fecho de Sockets (ms)	0	0	0	0	0	0	0%
Criação de Mensagens (ms)	538	1086	2906	5976	12332	4568	29.30%
Geração e Inserção de Dados DB-A (ms)	389	803	1971	3971	7976	3022	19.38%
Processamento de mensagem no Sistema B (ms)	998	1980	4783	10058	22180	8000	51.32%
Tempo de execução total (ms)	1925	3869	9660	20005	42489	15590	100%
Número de mensagens enviadas	50	100	250	500	1000	380	-
Tempo médio por mensagem enviada (ms)	38.5	38.69	38.64	40.01	42.49	41.07	-

Tabela 2: Envio batch com N aberturas TCP

3.3 1 Abertura de Sockets TCP e Envio Sequencial

	Teste #1	Teste #2	Teste #3	Teste #4	Teste #5	Média	Média Percentual
Abertura de Sockets (ms)	0	0	0	0	0	0	0%
Fecho de Sockets (ms)	0	0	0	0	0	0	0%
Criação de Mensagens (ms)	11570	11384	11386	11400	14349	12018	20.07%
Geração e Inserção de Dados DB-A (ms)	21075	21275	21207	21251	19338	20830	34.78%
Processamento de mensagem no Sistema B (ms)	27228	27216	27294	27185	26289	27042	45.15%
Tempo de execução total (ms)	59873	59875	59887	59836	59976	59890	100%
Número de mensagens enviadas	1410	1353	1369	1363	1154	1330	-
Tempo médio por mensagem enviada (ms)	42.46	44.25	43.75	43.71	51.98	45.03	-

Tabela 3: Envio sequencial com N aberturas TCP

3.4 1 Abertura de Sockets TCP e Envio Batch

	Teste #1	Teste #2	Teste #3	Teste #4	Teste #5	Média	Média Percentual
Abertura de Sockets (ms)	0	0	0	0	1	0	0%
Fecho de Sockets (ms)	0	0	0	0	0	0	0%
Criação de Mensagens (ms)	643	1275	3182	6844	14195	5228	32.45%
Geração e Inserção de Dados DB-A (ms)	403	810	2018	4089	8156	3096	19.21%
Processamento de mensagem no Sistema B (ms)	939	1869	5034	10376	20715	7787	48.34%
Tempo de execução total (ms)	1985	3954	10234	21309	43066	16110	100%
Número de mensagens enviadas	50	100	250	500	1000	380	-
Tempo médio por mensagem enviada (ms)	39.70	39.54	40.94	42.62	43.07	42.40	-

Tabela 4: Envio batch com N aberturas TCP

3.5 Compilação dos Resultados Obtidos

Como podemos constatar pelas tabelas anteriores, as tarefas de *Abertura do Socket* e do *Fecho do Socket* ocuparam sempre uma média temporal de 0%, logo, não serão mencionadas na compilação dos resultados. Para propósito de simplificação, os testes realizados serão referidos na seguinte tabela, da seguinte forma:

- N Aberturas de Sockets TCP e Envio Sequencial - **Teste A**
- N Aberturas de Sockets TCP e Envio Batch - **Teste B**
- 1 Abertura de Sockets TCP e Envio Sequencial - **Teste C**
- 1 Abertura de Sockets TCP e Envio Batch - **Teste D**

A partir dos resultados obtidos anteriormente, foi-nos possível agregá-los de modo a obter uma melhor compreensão do período temporal que cada tarefa demora a ser executada, como podemos confirmar na tabela abaixo:

	Teste A	Teste B	Teste C	Teste D	Média	Média Per- centual
Criação de Mensagens (ms)	11825	4568	12018	5228	8410	22.24%
Geração e Inserção de Dados DB-A (ms)	22375	3022	20830	3096	12331	32.62%
Processamento de mensagem no Sistema B (ms)	25439	8000	27042	7787	17067	45.14%
Tempo de execução total (ms)	59639	15590	59890	16110	37808	100%
Número de mensagens enviadas	1265	380	1330	380	839	-
Tempo médio por mensagem enviada (ms)	47.15	41.03	45.03	42.40	45.06	-

Tabela 5: Compilação de Resultados Obtidos

A partir desta tabela, podemos tirar mais facilmente conclusões em relação ao estudo feito, como por exemplo:

1. as tarefas seguem sempre a mesma tendência, demorando mais ou menos a mesma fatia temporal. A tarefa do *Processamento de Mensagens no Sistema B* é sempre a mais demorada, a *Geração e Inserção de Dados na DB-A* a segunda mais demorada e a *Criação de Mensagens* é sempre a mais célere
2. os testes que envolvem o envio das mensagens em batch, apresentam um envio de uma mensagem mais rápido em média do que o envio sequencial das mensagens, em cerca de 3ms, o que resulta num envio mais eficiente em cerca de 6.67%

3.6 Análise dos Resultados Obtidos

Se considerarmos o *Envio Sequencial com N aberturas de Sockets TCP* o **cenário base**, com os testes efetuados, conseguimos atingir, como foi mencionado anteriormente uma melhoria dos resultados obtidos em cerca de 6.67%.

Existem algumas razões para esta melhoria ser tão pequena, mas a principal razão é o facto de as tarefas que podem ser efetuadas apenas uma vez, como a abertura / fecho dos sockets TCP, não são significativas no tempo final de execução, e as tarefas que realmente demoram, não são facilmente efetuadas concorrentemente.

3.7 Possíveis Melhorias

Apesar de serem testados alguns cenários, não foi possível efetuar testes a todos os fatores que influenciam o envio das mensagens HL7, e como tal, gostaríamos de sugerir alguns tópicos que poderiam ser estudados futuramente quanto ao seu impacto numa experiência semelhante:

1. **Envio concorrente de mensagens HL7** - A concorrência sempre foi um vantagem no mundo informática para possibilitar a execução mais eficiente de tarefas e esta poderia ser aplicada cá, por exemplo, do lado do Sistema B, de modo a processar N mensagens HL7 simultaneamente, reduzindo a fatia temporal da tarefa mais demorada neste processo todo. É óbvio que é necessária a reunião de algumas condições para que isto seja possível, como por exemplo os recursos computacionais, mas assumindo que essas condições são atingidas, isto poderia vir a melhorar significativamente o desempenho do sistema.
2. **Implementação de um Gerador de Mensagens HL7 Mais Eficiente** - Neste exercício, recorremos a uma biblioteca externa que nos possibilitava a criação de mensagens HL7. A utilização de bibliotecas desconhecidas, regularmente traz algum overhead, de funcionalidades desnecessárias que podem vir a impactar negativamente o desempenho do sistema como um todo. Uma solução possível, poderia ser a implementação de uma biblioteca própria, sem funcionalidades que não seriam utilizadas, de modo a tornar esta tarefa o mais eficiente possível.

4 Conclusões

Através deste exercício prático fomos capazes de criar um sistema de envio e receção de mensagens, permitindo a avaliação do desempenho do mesmo, através de estudos estatísticos, recorrendo a médias de tempos de envio por mensagem, totais de mensagens enviadas, entre outras métricas.

Graças a este estudo, fomos capazes de determinar onde se encontravam os *bottlenecks* no sistema desenvolvido, podendo ainda determinar soluções possíveis para estes problemas. Prevemos também alguns esforços futuros que poderiam também estes ser estudados, num trabalho futuro.

Para finalizar, cremos que este trabalho foi uma mais-valia, já que nos deu uma boa introdução aos estudos de complexidade envolvendo sistemas informáticos, neste caso, na geração e envio de mensagens HL7, no campo médico.

Referências

- [1] Python:
newblock<https://www.python.org/>
- [2] hl7apy:
<https://pypi.org/project/hl7apy/>
- [3] MySQL:
<https://www.mysql.com/>