

Experimental Performance Evaluation of Data-Distribution Service software

Arkadiusz Nowacki

May 2023

Introduction

Background of the project topic

The field of Data Distribution Services was initially presented by the company named Real-Time Innovations [1] (RTI) back in 2001 when the company started their development of DDS specification together with Thales Group. In the meantime, another group called Object Management Group (OMG) worked on their version of the technology in 2004 and continued to release newer versions in 2005, 2007 and 2015 until it was accepted by the US patent scheme. In the time of connectivity evolution, simple systems required simple links of connection between computers. The concept of scalability was limited to the increase in the number of communication links [2] between the continually added machines, which ultimately failed once there were enough machines, making the communication links become a significant part of the mechanism. In addition to that situation, they have started to dominate these systems, so a new concept has appeared to add a central broker that manages the communication between the nodes. However, it also was not practical in the long run because of the disadvantages it has introduced, such as a single point of failure, additional network latency based on the broker's load, and maintenance of new elements in the mechanism. DDS was a direct answer to avoiding these issues by the use of a new topology that would bring benefits while being a reliable choice across many applications/systems. Yet there still exists a gap in the field of DDS where there is no solution stating what company has produced the best solution to date. We cannot determine what DDS is the best, and based on what basis it would be advantageous more than the others. The most problematic point of this issue is an absence of a source of information that would encapsulate all of this data into one concrete place.

Project's purpose and product

The purpose of this project is to perform testing with a researched benchmark application that was pointedly designed for the needs of one company in the Data Distribution Service technology against the solutions developed by the various other vendors available. The reasoning is that the same benchmark application can be repurposed or enhanced to make it deployable and non-firm-specific. Conventionally, such performance test has a plugin architecture, which forges such possibility [9]. At the moment, the set of measurements of the non-security-enabled DDS in the form of latency and throughput should be the minimum output of the project, but subsidiary data could expand to a point in which the configuration of DDS security was enabled and measured as well. Naturally, these measurements would be collected, evaluated and processed for multiple products and in mentioned configurations within an experimental frame of this venture so that comparisons and findings between DDS products are viable. On the other hand, the report is going to be prepared at the end of the project plan to make a central point of such information.

Aims

The project aims to experimentally evaluate the performance of a set of DDS software products and compare their results.

Contribution

The contribution coming from the knowledge produced at the end of the experimental testing campaign for the Data Distribution Service will benefit groups and people such as:

- Individual programmers that create an application(s) in a data-centric manner want to maximize results in a specific metric. Based on knowing which solution is the best for their desired metric, they can make an informed choice and go to the appropriate vendor to use the code available for their work.
- Research teams within the DDS companies would benefit from knowing the performance evaluation details so they can enhance their research with a new piece of data.
- Individual users of the DDS technology want to be informed of the current status of available DDS implementations before they make their own choice. They may want to compare the market options before going to a further stage with any vendor.
- People interested in the batches of data, like data scientists and hobbyists.
- Internet of Things (IoT) businesses that want to choose an available DDS solution which minimizes latency better than other companies in the Data Distribution Service space so that the users of their technology, like voice assistants, can get the response to a request in the shortest time.
- Aerospace and Defence businesses that require high throughput and quick response time for tracking technologies ranging from satellites to radars.
- Automation businesses ranging from transport, robotics and autonomous vehicles require low-latency data connectivity, extreme reliability, and scalable architecture. They will benefit from the project's output and perhaps improve or adjust their choice in the Data Distribution Service.

Objectives

The objectives of the project are:

- To find and research a benchmark application that can be repurposed or enhanced to deploy it against at least two other Data Distribution Services without Security-enabled features.
- To produce experimental measurements for the Data Distribution Services of choice that are analysable, evaluable and subjected to comparison while encapsulated in a report at the end of the project.
- If possible, use the benchmark application on Data Distribution Services while their additional security configuration is enabled to expand the findings.

Critical Context

Data Distribution Service

The Data Distribution Services are essentially middleware protocols that can also include API standards for time and task-critical applications that are often data-centric in their nature [10]. The most popular and available options are eProsima Fast DDS, RTI DDS, Adlink Vortex

OpenSplice DDS, TwinkOaks Computing CoreDX DDS and OpenDDS. They preserve the Quality of Service policy in their communication because it relates to the nature of their usage. The definition proposed by OMG can also be expanded with more QoS specifications, which include reliability, system health and security [3]. The concept of data-centricity had an impact because of expansion from the familiar pathway programmers have chosen for middleware, which would send information between applications and systems to a system called message-centric middleware. The utilisation of data-centricity allows the code to be about how and when the data share should occur and then directly performs the value sharing. More importantly, the situation explained before happens from the DDS that controls, manages and secures data sharing for the user and obscures such complexity from the application's code. In simple terms, the DDS with QoS-controlled data sharing follows a new topology in which the applications communicate via publishing and subscribing to "Topics" uniquely identified to make communication predictable and linear. The idea behind topics is to generate virtual data transmission channels [10] between the publisher and subscriber, where once the publishers generate some new data for a topic, all its content goes directly to subscribers of this topic in a DDS local store of data called "Global Data Space". Once we follow each documentation of the DDS implementation provider, there is a clear definition of the same principle. The point of difference is naming conventions. For example, the global data space may change to DDS Databus for RTI [2] and Data Transmission for OpenDDS [5]. Based on all the data collected in this section, a better understanding of the project space has been gathered with a search through the available options in the market to identify the spaces of further investigation as the project plan begins.

Performance tests done by DDS companies

Every time the business or individual wants to use the standard Data Distribution Service will come across a limited number of commercial and open-source implementations available with various information about their performance in latency and throughput (sometimes with additional information such as CPU usage). The best-known companies and their implementation of the standard are eProsima Fast DDS, RTI DDS, OpenSplice DDS, CoreDX and OpenDDS [4]. It is a frequent business practice to have a page dedicated to the results of the already well-established technology the company offers. Once we get a page dedicated to such results provided by the eProsima about their DDS implementation, there is information about the general latency and throughput as well as the advanced based on mechanisms measurements. The values in "general latency test" [7] are encapsulated in a median in the range of 10-20 μ s across payload in bytes starting from 0 and increasing by 5 thousand until the limit of 15,000. The natural step is to compare to a competitor, which in this case is RTI, to clarify the issue. The problem is that this company provides "one-reliable latency" [8] that is not a median in the " μ s" units, as well as the number of payloads in bytes, happens to be in a different range (starting from 0, 32, 64, 128 ... with a gap until 63 thousand). Such disparity between the results forces the business and individuals to spend time and perhaps money to convert the metrics and then make an analysis. Complementary to this, the effort could increase while the number of comparisons becomes higher as someone may want to know the best solution currently available in the market from a range of offers. Then, the actions mentioned may exclude individuals that do not possess the same capabilities to perform analysis. Nevertheless, the problem of putting measurements of more than one company together is impossible due to several factors that have affected the results, which makes them incomparable. These factors could include but are not limited to the date

of the test, software, hardware, methodology, configuration, choices of what to measure, and polishing of these tests specifically to the company's DDS implementation. An efficient approach to getting a truthful comparison in most technology available is to preserve the same environment in which the testing campaign happens. Although all of the listed DDS vendors have their performance testing page, none has constraints by a global standard of testing that would outline how the tests ought to be performed. Due to that implication, we can observe the use of different hardware and software by CoreDX DDS [14] and OpenSplice DDS [15] such as the switch provider Netgear and Dell, as well as not matching versions of the Linux operating system, where one is "Linux 3.8.13-rt14.20.el6rt.x86_64" and other "Linux 2.6 x86" in the testing machines. Consideration of all of this context gives a clear idea that for experimental performance evaluation, the benchmarking application should be independent of these factors and produce cohesive, comparable and informative output once used. In conclusion, now it is known that there is a clear path that the project should take and informed choices can be made to avoid aspects that could harshly affect the project's output.

Benchmark application

Depending on the provider of the Data Distribution Service, there is a limit to having access to a combined latency and throughput test application that could potentially work for more than one DDS implementation. Based on the comment exchange that a student had with the RTI employee Howard Wang we can see some third-party options and the company-specific solution [9]. The performance test associated with each offer was clearly outlined, whereas only specific entities allow a full view of the source code and tutorials on performing tests on our own. Consideration of the performance test options that will be rewritten or repurposed for the testing campaign is, therefore, a vital step of the project so that there is a base knowledge before the more in-depth literature review.

RTI PerfTest

RTI PerfTest [11] is a fully documented latency and throughput measuring software available as an open-source project on a GitHub repository. Once the communication is established between the perfTest subscriber and publisher, the publishing side writes the data as quickly as possible based on the configuration. Such configuration also specifies that every few samples are responses called "echos". They are sent to the opposite side, so the so-called round-trip time latency can be calculated (and divided by two). On the other hand, there is another option to run multiple copies of the application where the person testing can adjust the number of subscribers while usually keeping one publisher. The publishing application will publish throughput data while subscribing to the latency responses. Accordingly, the subscribing applications subscribe to throughput data where the echos samples are placed while publishing latency echos. Based on these operations the test allows retrieval of latency test results through the publisher and throughput test results through the subscriber. The evidence above suggests a simple definition of perfTest and outlines a structure in which the tests are executable in the same way as proving that the perfTest is a candidate to be used to meet the purpose of the project.

DDS Benchmark Environment

DBE is a DDS benchmarking environment that can be executed against different implementations of DDS by utilizing Perl scripts for automotive tests on a distributed network [12]. This benchmark software was once technically backed by the RTI and could be run by anyone through a secure shell to launch publishers and subscribers and coordinate

interaction while logging information. coordinate interaction while logging information. Its versatility allowed it to measure throughput, CPU usage, and memory, although no mention of measuring the latency on the main page or PowerPoint presentation would suggest otherwise. At the moment, the scripts are not available from the documentation page of the tool, but they can be found with a bit of navigation. The main issue of this test is that its outdated for a very long time, which suggests the project was abandoned and can only act as a good reference in the creation/adaptation of the up-to-date choice.

DDSBench

DDSBench was developed by the employer of the PrismTech company, which is a subsidiary company under Adlink that offers OpenSplice DDS in the form of an executable project written in C language. The project itself does not have any documentation outside of the comments in the code, but the portions of the code in the "throughput" and "roundtrip" [13] files relate to operations performed in the perftest following the same principle between subscribers and publishers to measure the latency and throughput. In the same way, as in the previous case, the source gives a portion of knowledge about how experimental performance evaluation can be done as the project goes into the development stage. Despite it being an outdated source, it also validates the need for the existence of the project, similar to the last environment.

Approaches: Methods & Tools for Design, Analysis & Evaluation

Literature Survey

The literature search started with the initial supervisor meeting, where a list of DDS was discussed. Secondly, I have reviewed their document/blog/tutorial/performance pages to determine the project's purpose and objectives. However, some time should be allowed to find out if other data distribution services offer better value in terms of benchmarking tools. There needs to be a more precise investigation into what current performance tests exist and whether they can be used against more than one DDS implementation similar to what is supposedly possible with the perftest.

Analysis and Evaluation

To make the experiment measurements provided by the benchmark software practical, where it was collected across more than one company, it needs to generate output in the form of numerical batches of data. The values should be in a form such that they can be analysed and evaluated even when they come from the different implementations of the DDS software, as it is necessary to establish leadership or the point at which one company's DDS is more advantageous than the other.

To evaluate the performance of the DDS implementations our experiment can use a standard table format to keep a quantitative record of the latency and throughput. For each implementation, the table can consist of values not limited to only medians, as it was common practice mentioned in section "Performance tests done by DDS companies". The values in a table(s) would consider latency, throughput [11] and perhaps other measurements such as CPU and memory utilisation [6]. For each table of the given independent DDS product under the given configuration of security (enabled/disabled), the data batches would be subjected to qualitative data analysis with descriptive statistics that will result in the extraction of most likely the average, median, mode (where last three are central

tendency/average information), minimum, and maximum values. Moreover, a suitable choice in the tables of values could be the cumulative distribution function (CDF) which gives the probability that random variable "X" is less than or equal to "x" in accordance to function $F(x)=P[X \leq x]$ and return the result as a value between 0 and 1. In this case, we would include the inferential statistic to enhance our observations through predictions. The possibility comes from the function being definable for any random variable appearing in the data batches. It will also detail information about the software and hardware, as well as the time frame in which the possible tests will be carried out to ensure that everything is as clear as possible in the outcome of the project. While adding additional values to the tables increases the amount/sophistication level of information and the possibilities of judgment, they may be subjected to change upon unforeseen circumstances/obstacles of the benchmarking software. Nevertheless, the products of the experiment will certainly be able to indicate the best option in terms of the measurements an individual or business is interested in.

To improve the process of analysis several graphical visualisations will likely be included in the form of presenting connected data markers on the x and y axis, which we might collect over time when the data batches come when the experimental testing campaign happens. A step before the visualisations would be to process the data in a way that perhaps excludes outlier values so that the clean data can be considered. It would allow the research process to be more efficient with additional sampling choices such as random sampling or own-choice sampling where a specific window of data is selected, for example, starting to the middle point of all values. The likely advantage of doing that is not being affected by raising errors or bias. Moreover, consideration of the limit arises from the fact that it is not known how much data is being generated then, so this may affect how the data is presented. Secondly, similarly to the previous way of data presentation, it might be helpful to make a conclusive scatter plot that would display data points and a histogram that would show the frequency of latency and throughput values. A choice of representation based on previous information can also include the CDF. The result of the proposed methods would enrich the findings from the data batches that most likely would not be considered in the tables.

Ultimately, there will be an assessment of how each DDS software has performed based on the latency and throughput (possibly with more metrics) through direct A/B comparisons so that the best-performing software under specific measurement can be distinguished with the additional specification of how it was configured. All of the rankings would be presented using a graphical form of a table or/and graph indicating the strengths of the implementation in a file or the number of files that will get to the final report.

Software

The first step will be to choose software for the benchmark application that will be used for the project upon finishing the literature survey at the beginning of the project. Once the development of enhancement of existing solutions will start in the middle of the work plan a number of applications will be used such as IntelliJ integrated development environment for code writing while controlling the versions of the software with Git to work with the existing solution's code.

Another software used will be Microsoft Excel which allows the data collected to be stored in a popular file format such as ".csv", ".txt", and ".xlsx" making it possible to have well-

structured and documented tables of results. More importantly, allowing data pre-processing, clearing and plotting with Python libraries such as matplotlib, pandas and numpy inside of web interactive computing platforms such as Jupyter Notebook or Google Colab. They provide the necessary tools but in case of expansion or collection of more data than was expected other choices such as MATLAB are reachable.

Project meetings

The project meetings are currently set for every Thursday at around 5 pm (UK time) when additional support is required with either source or clarification of a concept(s). It helps the project to stay in the lane while allowing informative changes in the work plan if necessary. They are not likely to be done each week, but it is possible to have them once it is necessary and established with the project supervisor. When the project will not require communication of this sort, the other communication will be based on updates via email about the current progress and obstacles.

Ethical, Legal and Professional Issues

Based on what is stated in the ethics form there will not be the participation of any third parties. The work is produced with the use of literature and multiple online sources. The material used in this work is cited and outlined in each part of the work.

Work Plan

Tasks	Phase	Time
1	Literature Survey to search for the existing benchmarking solutions and review their documentation/code/use cases/possibilities.	3 weeks (1 st of June to 28 th of June)
2	Going through the reviewed benchmark application choices and deciding on which of them will be used to run a test campaign in the future.	1 week (28 th of June to 5 th of July)
3	Repurposing or enhancing benchmark application for other DDS products. It means that the structure of a plugin has to be developed in a way that works for other DDS solutions, which will be also checked in this development stage.	5 weeks (5 th of July to 9 th of August)
4	Running the testing campaign against DDS products (with security configuration disabled and, if possible, enabled).	2 weeks (9 th of August to 23 rd of August)
5	Collection of data from the tests as the measurements of latency and throughput as the batches of data.	1 week (23 rd of August to 30 th of August)
6	Postprocessing of the data, putting data into a tabular format and the graphs, and then running comparisons between the results and ranking them accordingly.	2 weeks (30 th of August to 13 th of September)
7	Finalising the project report.	2 weeks 5 days (13 th of September to 2 nd of October)

Risks

Risk	Likelihood (1-3)	Consequences (1-5)	Impact (L x C)	Mitigation Strategy
Misunderstanding parts of the source-code provided for the benchmark application.	1	2	2	Spend more time on the review of the code, identify similarities and strategies and look through the comments or the "README" sections as well as the documentation pages to have a full understanding before more code is written or reused.
Measuring the performance of DDS software and forgetting whether security configuration is enabled or disabled at the time of testing.	2	5	10	For the testing campaign, start with the security configuration disabled for all the DDS software and collect these results and then run the second half of the tests with the option enabled. Whether the result is written to a file or software, including an indication of which result is security enabled/disabled with words like "security on" and "security off" or "on" and "of" for quick reference.
DDS software does not want to work with benchmarking application.	3	4	12	In Task 3, make sure to choose the DDS software that is less likely to have a problem working with benchmarking applications, has more extensive documentation and is more popular than others, so there exists a point of contact with the company or its community. It could also include working with DDS software that has similar syntax/principles in its source code. Lastly, a backup choice should be considered in case such a situation happens to minimize the likelihood.
Batches of data are too large to analyse and evaluate.	2	3	6	Choose only a portion of the data and adapt the same strategy of choosing it across all the DDS data collection to establish a common approach.
DDS benchmark application source code is unavailable for a view.	2	4	8	Send an e-mail to the company's representative or a person actively committing their code (this person could be identified via GitHub) to obtain in or ask questions that can help without the code.
Code was lost or overwritten.	2	5	10	Go to the Git version control and find the code that was lost or overwritten.
Postprocessing, putting data into tabular/graph formats, takes much longer than it is supposed to.	1	3	3	Make the data in a tabular format ready and then go to graph data and reduce the number of graphs that were planned to be done to only essential and most informative for the beneficiaries.

References

- [1] *wikipedia.org* (2022). Data Distribution Service. [online] Available at: https://en.wikipedia.org/wiki/Data_Distribution_Service (Accessed 09 May 2023)
- [2] *rti.com* (n.d.). Data Distribution Service (DDS) for Complex Systems — RTI. [online] www.rti.com. Available at: <https://www.rti.com/products/dds-standard> (Accessed 09 May 2023)
- [3] *ddsfoundation.org* (n.d.). What is DDS? [online] Available at: <https://www.ddsfoundation.org/what-is-dds-3/> (Accessed 18 April 2023)
- [4] *eprosima.com* (n.d.). DDS. [online] Available at: <https://www.eprosima.com/index.php/resources-all/whitepapers/dds> (Accessed 19 April 2023)
- [5] *opendds.readthedocs.io* (n.d.). Introduction - OpenDDS 3.24.1. [online] Available at: <https://opendds.readthedocs.io/en/dds-3.24.1/devguide/introduction.html#publisher> (Accessed 18 April 2023)
- [6] *opendds.readthedocs.io* (n.d.). Bench Performance & Scalability Test Framework - OpenDDS 3.24.1. [online] Available at: <https://opendds.readthedocs.io/en/dds-3.24.1/internal/bench.html> (Accessed 19 April 2023)
- [7] *eprosima.com* (n.d.). eProsima Fast DDS Performance. [online] Available at: https://www.eprosima.com/index.php/resources-all/performance/eprosima-fastdds-performance#General_Latency_Comparison (Accessed 09 May 2023)
- [8] *rti.com* (n.d.). Connex Performance Benchmarks—RTI Products. [online] www.rti.com. Available at: <https://www.rti.com/products/performance> (Accessed 25 April 2023)
- [9] *rti.com* (n.d.). Is There a Benchmarking Tool That Tests the Performance of Multiple DDS Products As Well As RTI — Data Distribution Service (DDS) Community RTI Connex Users. [online] Available at: <https://community.rti.com/forum-topic/there-benchmarking-tool-tests-performance-multiple-dds-products-well-rti> (Accessed 25 April 2023)
- [10] *dx.doi.org* (n.d.). Sim W, Song B, Shin J and Kim T 2021 Data Distribution Service Converter Based on the Open Platform Communications Unified Architecture Publish–Subscribe Protocol Electronics 10 2524 [online] Available at: <http://dx.doi.org/10.3390/electronics10202524> (Accessed 18 April 2023)
- [11] *community.rti.com* (n.d.). 1. Introduction — RTI PerfTest 4.1 documentation. [online] Available at: <https://community.rti.com/static/documentation/perftest/current/introduction.html#> (Accessed 7 May 2023)
- [12] *dre.vanderbilt.edu* (n.d.). DDS Benchmark Environment. [online] Available at: <https://www.dre.vanderbilt.edu/DDS/html/dbe.html> (Accessed 7 May 2023)
- [13] *github.com* (n.d.). ddsbench/src at master · cwyark/ddsbench. [online] Available at: <https://github.com/cwyark/ddsbench/tree/master/src> (Accessed 28 April 2023)
- [14] *twinoakscomputing.com* (n.d.). Performance: Network Speed — Twin Oaks Computing, Inc. [online] Available at: <https://www.twinoakscomputing.com/coredx/performance-network> (Accessed 27 April 2023)
- [15] *adlinktech.com* (n.d.). Vortex OpenSplice - DDS Performance - ADLINK. [online] ADLINK Technology. Available at: <https://www.adlinktech.com/en/vortexopensplice-performance> (Accessed 27 April 2023)

Research Ethics Review Form: BSc, MSc and MA Projects

Computer Science Research Ethics Committee (CSREC)

A.1 If you answer YES to any of the questions in this block, you must apply to an appropriate external ethics committee for approval and log this approval as an External Application through Research Ethics Online - https://ethics.city.ac.uk/		<i>Delete as appropriate</i>
1.1	<p>Does your research require approval from the National Research Ethics Service (NRES)?</p> <p><i>e.g. because you are recruiting current NHS patients or staff?</i></p> <p>If you are unsure try - https://www.hra.nhs.uk/approvals-amendments/what-approvals-do-i-need/</p>	NO
1.2	<p>Will you recruit participants who fall under the auspices of the Mental Capacity Act?</p> <p><i>Such research needs to be approved by an external ethics committee such as NRES or the Social Care Research Ethics Committee - http://www.scie.org.uk/research/ethics-committee/</i></p>	NO
1.3	<p>Will you recruit any participants who are currently under the auspices of the Criminal Justice System, for example, but not limited to, people on remand, prisoners and those on probation?</p> <p><i>Such research needs to be authorised by the ethics approval system of the National Offender Management Service.</i></p>	NO
A.2 If you answer YES to any of the questions in this block, then unless you are applying to an external ethics committee, you must apply for approval from the Senate Research Ethics Committee (SREC) through Research Ethics Online - https://ethics.city.ac.uk/		<i>Delete as appropriate</i>
2.1	<p>Does your research involve participants who are unable to give informed consent?</p> <p><i>For example, but not limited to, people who may have a degree of learning disability or mental health problem, that means they are unable to make an informed decision on their own behalf.</i></p>	NO
2.2	<p>Is there a risk that your research might lead to disclosures from participants concerning their involvement in illegal activities?</p>	NO
2.3	<p>Is there a risk that obscene and or illegal material may need to be accessed for your research study (including online content and other material)?</p>	NO
2.4	<p>Does your project involve participants disclosing information about special category or sensitive subjects?</p> <p><i>For example, but not limited to: racial or ethnic origin; political opinions; religious beliefs; trade union membership; physical or mental health; sexual life; criminal offences and proceedings</i></p>	NO
2.5	<p>Does your research involve you travelling to another country outside of the UK, where the Foreign & Commonwealth Office has issued a travel warning that affects the area in which you will study?</p>	NO

	Please check the latest guidance from the FCO - http://www.fco.gov.uk/en/	
2.6	Does your research involve invasive or intrusive procedures? <i>These may include, but are not limited to, electrical stimulation, heat, cold or bruising.</i>	NO
2.7	Does your research involve animals?	NO
2.8	Does your research involve the administration of drugs, placebos or other substances to study participants?	NO
A.3 If you answer YES to any of the questions in this block, then unless you are applying to an external ethics committee or the SREC, you must apply for approval from the Computer Science Research Ethics Committee (CSREC) through Research Ethics Online - https://ethics.city.ac.uk/ Depending on the level of risk associated with your application, it may be referred to the Senate Research Ethics Committee.		Delete as appropriate
3.1	Does your research involve participants who are under the age of 18?	NO
3.2	Does your research involve adults who are vulnerable because of their social, psychological or medical circumstances (vulnerable adults)? <i>This includes adults with cognitive and / or learning disabilities, adults with physical disabilities and older people.</i>	NO
3.3	Are participants recruited because they are staff or students of City, University of London? <i>For example, students studying on a particular course or module.</i> <i>If yes, then approval is also required from the Head of Department or Programme Director.</i>	NO
3.4	Does your research involve intentional deception of participants?	NO
3.5	Does your research involve participants taking part without their informed consent?	NO
3.5	Is the risk posed to participants greater than that in normal working life?	NO
3.7	Is the risk posed to you, the researcher(s), greater than that in normal working life?	NO
A.4 If you answer YES to the following question and your answers to all other questions in sections A1, A2 and A3 are NO, then your project is deemed to be of MINIMAL RISK. If this is the case, then you can apply for approval through your supervisor under PROPORTIONATE REVIEW. You do so by completing PART B of this form. If you have answered NO to all questions on this form, then your project does not require ethical approval. You should submit and retain this form as evidence of this.		Delete as appropriate
4	Does your project involve human participants or their identifiable personal data? <i>For example, as interviewees, respondents to a survey or participants in testing.</i>	NO