Добрый день, уважаемые коллеги!

Данным письмом я открываю серию писем, связанным с организацией создания информационных систем и разработкой программного обеспечения в отделе 08402 в условиях импортозамещения, а также с организацией процесса разработки программного обеспечения с использованием отечественных средств разработки.

1. **Архитектура**

При создании информационных систем и разработке программного обеспечения мы традиционно придерживались трехуровневой архитектуры, в которой задействовали систему управления базами данных, сервер приложений (web-сервер) и клиентские рабочие места, использующие web-браузер в качестве клиента.

Мы будем продолжать придерживаться данной архитектуры и в новых условиях. В некоторых случаях возможно так же использование двухуровневой клиент-серверной архитектуры.

1. **Аппаратное обеспечение**

Мы традиционны использовали зарубежное аппаратное обеспечение (сервера, системы хранения данных, принтеры, рабочие станции, сетевые устройства).

В настоящее время на рынке достаточно широко представлено отечественное аппаратное обеспечение, производимое различными российскими компаниями. Мы будем использовать аппаратное обеспечение, включенное в Реестр промышленной продукции, произведенной на территории Российской Федерации Министерства промышленности и торговли Российской Федерации, доступного по ссылке <https://gisp.gov.ru/pp719v2/pub/prod/>

Как пример – сервера (<https://www.depo.ru/catalog/servery/servery-iz-reestra-minpromtorga-rf/>), рабочие станции (<https://www.depo.ru/catalog/kompyutery/kompyutery-iz-reestra-minpromtorga-rf/>) и системы хранения данных (<https://www.depo.ru/catalog/sistemy-khraneniya-dannykh/>) от отечественного производителя Depo Computers.

1. **Программное обеспечение**

Мы традиционны использовали программное обеспечение импортного происхождения. В новых условиях нам необходимо использовать отечественное программное обеспечение из «Единого реестра российских программ для электронных вычислительных машин и баз данных», доступного по ссылке <https://reestr.digital.gov.ru/>

Далее рассматривается программное обеспечение, которое мы будем использовать для развертывания разрабатываемого нами программного обеспечения.

* 1. **Система виртуализации**

В настоящее время мы используем систему виртуализации VMWare vSphere.

В дальнейшем мы будем использовать отечественные системы виртуализации «Р-Виртуализация» (<https://rosplatforma.ru/product/virtualizatsiya/>, <https://reestr.digital.gov.ru/reestr/304698/>) и zVirt (<https://orionsoft.ru/zvirt>, <https://reestr.digital.gov.ru/reestr/306334/?sphrase_id=1658630>).

Систему виртуализации «Р-Виртуализация» рекомендуется к использованию при создании информационных систем, не имеющих в своем составе выделенной системы хранения данных. В этом случае систему виртуализации рекомендуется использовать совместно с системой распределенного хранения данных «Р-Хранилище» (<https://rosplatforma.ru/product/khranilishche/>, <https://reestr.digital.gov.ru/reestr/304730/>) для обеспечения распределенного избыточного хранения данных с целью обеспечения надежного хранения и отказоустойчивости.

Систему виртуализации zVirt рекомендуется к использованию при создании информационных систем, имеющих в своем составе выделенную систему хранения данных.

* 1. **Операционная система серверов**

В настоящее время мы используем операционные системы Oracle Linux (бинарный аналог операционной системы Red Hat Linux) и Microsoft Windows Server.

В дальнейшем мы будем использовать операционную систему Astra Linux Special Edition (бинарный аналог операционной системы Debian Linux, <https://astralinux.ru/products/astra-linux-special-edition/>, <https://reestr.digital.gov.ru/reestr/301718/>).

Данная операционная система выпускается в трех уровнях защищенности – базовом, усиленном и максимальном. Поскольку в большинстве случаев перед нами не стоит задача обработки конфиденциальной или секретной информации, мы будем использовать операционную систему в базовом уровне защищенности.

* 1. **Система управления базами данных**

В настоящее время мы используем СУБД Oracle Database в различных версиях и конфигурациях.

В дальнейшем мы будем использовать СУБД PostgreSQL (<https://ru.wikipedia.org/wiki/PostgreSQL>), входящую в состав операционной системы Astra Linux Special Edition. Поскольку версия PostgreSQL, входящая в состав операционной системы Astra Linux Special Edition не поддерживается разработчиком операционной системы и имеет значительное отставание по номеру версии и возможностям от версии СУБД PostgreSQL, выпускаемой сообществом разработчиков PostgreSQL, то для особо важных приложений нами будет использоваться отечественная СУБД Postgres Pro (<https://postgrespro.ru/products/postgrespro>, <https://reestr.digital.gov.ru/reestr/301574/?sphrase_id=1658777>)

* 1. **Виртуальная машина Java**

Поскольку для работы используемого нами сервера приложений необходимо наличие виртуальной машины Java, в настоящее время мы используем виртуальную машину Java производства компании Oracle.

В дальнейшем мы будем использовать отечественную виртуальную машину Java с названием Liberica JDK (aka Axiom JDK) (<https://axiomjdk.ru/>, <https://bell-sw.com/pages/downloads/>, <https://reestr.digital.gov.ru/reestr/306843/?sphrase_id=1658786>).

* 1. **Сервер приложений**

В настоящее время мы используем сервер приложений Oracle Web Logic Server.

В дальнейшем мы будем использовать сервер приложений LiberCat (<https://axiomjdk.ru/pages/libercat/>, <https://reestr.digital.gov.ru/reestr/310558/?sphrase_id=1658794>), созданный на основе проекта с открытым исходным кодом Apache Tomcat.

Из этого прямо следует, что программное обеспечение будет создаваться с использованием языка Java в соответствии со спецификациями Java EE.

* 1. **Web-сервер**

В настоящее время мы используем Web-сервер Oracle HTTP Server.

В дальнейшем мы будем использовать Web-сервер Apache HTTP Server, входящий в состав операционной системы Astra Linux Special Edition.

* 1. **Клиентская операционная система и web-браузер**

В настоящее время мы используем операционную систему Microsoft Windows 10 Professional с браузером Google Chrome.

В дальнейшем мы будем использовать операционную систему Astra Linux Special Edition (бинарный аналог операционной системы Debian Linux) с браузером Firefox, входящим в состав операционной системы.

* 1. **Офисные пакеты**

В настоящее время мы используем офисный пакет Microsoft Office. Для просмотра файлов PDF используется программное обеспечение Adobe Acrobat Reader или иное программное обеспечение для просмотра файлов PDF, доступное для операционной системы Microsoft Windows.

В дальнейшем мы будем использовать офисный пакет LibreOffice (<https://ru.wikipedia.org/wiki/LibreOffice>), входящий с состав операционной системы Astra Linux Special Edition. Для просмотра файлов PDF будет использоваться программа Qpdfview, входящая в состав операционной системы Astra Linux Special Edition.

1. **Резервное копирование**

Необходимо предусмотреть выполнение резервного копирования следующих компонентов информационных систем:

* образов виртуальных серверов;
* каталогов серверов, содержащих:
  + установленное общее и специальное программное обеспечение;
  + данные, обрабатываемые программным обеспечением;
  + настроечные и конфигурационные файлы общего и специального программного обеспечения;
* баз данных.

1. **Резервное копирование образов виртуальных серверов**

Резервное копирование образов виртуальных серверов должно выполнятся встроенными средствами системы виртуализации.

1. **Резервное копирование каталогов серверов**

Резервное копирование каталогов серверов может выполняться разными способами. Предпочтительным способом является организация в ЦУП централизованной системы резервного копирования. В этом случае в состав операционной системы вводится программный агент системы резервного копирования, который выполняет резервное копирование каталогов сервера по расписанию и передает полученные резервные копии для хранения в систему централизованного резервного копирования.

Так же возможно использование для выполнения этой операции встроенных утилит операционной системы Linux, например, rsync.

1. **Резервное копирование баз данных**

Резервное копирование баз данных должно выполняться в соответствии с руководством администратора для соответствующей базы данных, например, для PostgreSQL - <https://www.postgresql.org/docs/current/backup.html>.

При этом многие современные системы резервного копирования имеют в своем составе программные агенты, позволяющие обеспечить копирование баз данных по расписанию с последующей передачей полученных резервных копий для хранения в систему централизованного резервного копирования. В случае наличия такой системы резервного копирования целесообразно использовать ее.

1. **Стратегия резервного копирования и восстановления**

В обязательном порядке должна быть разработана и описана в документации на информационную систему стратегия ее резервного копирования и восстановления.

Стратегия резервного копирования и восстановления должна определять:

* порядок и частоту создания резервных копий (как полных, так и инкрементальных);
* количество одновременно хранимых резервных копий;
* порядок удаления хранимых резервных копий;
* порядок восстановления информационной системы из резервной копии;
* порядок и частоту выполнения тестовых восстановлений информационной системы из резервной копии.

Крайне рекомендуется проводить тестовые восстановления информационной системы из резервной копии не реже одного раза в год. Для проведения такого восстановления следует предусмотреть создание различных тестовых стендов.

1. **Наличие дистрибутивов общего и специального программного обеспечения**

При создании информационной системы следует обеспечить запись на носители однократной записи (CD-R, DVD-R) дистрибутивов установленного общего и специального программного обеспечения, нанести на носители и упаковку соответствующую маркировку и обеспечить учет и надежное хранение данных носителей.

1. **Обеспечение надежности и отказоустойчивости**

В настоящее время мы обеспечиваем работоспособность двумя способами.

Первый способ состоит в использовании опций High Availability и Fault Tolerance системы виртуализации VMWare vSphere. Опция High Availability обеспечивает немедленный перезапуск виртуального сервера в случае его сбоя или сбоя аппаратного обеспечения, на котором функционировал данный виртуальный сервер. Опция Fault Tolerance обеспечивает функционирование дополнительного теневого экземпляра основного виртуального сервера. Если происходит аппаратный сбой, то опция Fault Tolerance выполняет автоматический переход на теневой экземпляр виртуального сервера, делая его основным и обеспечивая, таким образом, нулевое время простоя. Данный метод используется для обеспечения надежности и отказоустойчивости серверов приложений, Web-серверов и СУБД.

Второй способ состоит в использовании кластерного программного обеспечения, например, Oracle Clusterware. Данное программное обеспечение используется для обеспечения работы СУБД Oracle Database в кластерной конфигурации, обеспечивающей работу кластера баз данных, состоящего из двух и более узлов, причем каждый узел работает в режиме Active (Oracle Real Application Cluster).

В дальнейшем мы планируем использовать опцию систем виртуализации «Р-Виртуализация» или zVirt, аналогичную опции High Availability, позволяющей обеспечивать немедленный перезапуск виртуального сервера в случае его сбоя или сбоя аппаратного обеспечения, на котором функционировал данный виртуальный сервер. В данном случае возможен перерыв в обслуживании на период времени, необходимого для осуществления перезапуска виртуального сервера. Опция, аналогичная опции Fault Tolerance системы виртуализации VMWare vSphere, в настоящее время отечественными системами виртуализации не поддерживается.

Так же и СУБД PostgreSQL не поддерживает работу в кластерной конфигурации, при которой каждый узел кластера работает в режиме Active. Мы в большинстве будем использовать кластер СУБД, состоящий из двух узлов, один из которых работает в активном (Active) режиме, а другой – в дежурном режиме (Passive). Так же возможно развертывание СУБД PostgreSQL на виртуальном сервере в конфигурации, состоящей из одного узла. В этом случае надежность и отказоустойчивость СУБД PostgreSQL обеспечивается средствами системы виртуализации, при этом возможно прерывание в работе СУБД на период времени, необходимого для осуществления перезапуска виртуального сервера.

1. **Обеспечение надежности и отказоустойчивости СУБД PostgreSQL**

Надежность и отказоустойчивость СУБД должна обеспечиваться созданием кластера, состоящего из двух узлов, один из которых работает в активном режиме (Active), а второй – в пассивном режиме (Passive). Между узлом, работающем в активном режиме, и узлом, работающем в пассивном режиме, организована репликация. Подробнее см. <https://www.postgresql.org/docs/current/high-availability.html>

При этом необходимо отметить, что СУБД PostgreSQL не обеспечивает автоматический переход с вышедшего из строя активного узла на пассивный узел и объявления его активным. Для такой операции необходимо использовать утилиты corosync/pacemaker (<https://pgconf.ru/media2015c/sontakke.pdf>, <https://www.youtube.com/watch?v=fOB49y2vGso>, <https://www.youtube.com/watch?v=eHAe323N994>).

Следует отметить, что отечественная СУБД Postgres Pro имеет более продвинутые средства обеспечения надежности и отказоустойчивости, которые нуждаются в дополнительном изучении (<https://www.youtube.com/watch?v=eHAe323N994>, с 34-ой минуты).

1. **Обеспечение надежности и отказоустойчивости сервера приложений**

Надежность и отказоустойчивость сервера приложений может обеспечиваться как системой виртуализации, так и встроенными механизмами сервера приложений, позволяющими создать кластер серверов приложений (<https://www.mulesoft.com/tcat/tomcat-clustering>).

1. **Обеспечение надежности и отказоустойчивости Web-сервера**

Надежность и отказоустойчивость Web-сервера может обеспечиваться как системой виртуализации, так и с использованием утилит corosync/pacemaker (<https://www.digitalocean.com/community/tutorials/how-to-set-up-an-apache-active-passive-cluster-using-pacemaker-on-centos-7>, <https://www.techmaze.be/#!/p/High%20Availability%20with%20Pacemaker%20and%20corosync>)

1. **Мониторинг**

В настоящее время мониторинг информационных систем осуществляется либо с использованием средства мониторинга Dell FogLight, либо вручную с использованием различных утилит из состава операционной системы или из состава используемого общего программного обеспечения. Для мониторинга аппаратной используется механизм iLO.

В дальнейшем мы планируем использование системы централизованного мониторинга Zabbix (<https://www.zabbix.com/>), уже развернутой и используемой в ЦУП. При использовании данной системы в состав операционной системы вводится программный агент (агент Zabbix), который осуществляет мониторинг и передает полученные события и метрики в централизованную систему мониторинга для отображения данной информации на специальных контрольных панелях и для организации оповещения пользователей о важных событиях. В состав операционной системы Astra Linux Special Edition включен агент Zabbix, настройку и подключение которого к централизованной системе мониторинга необходимо предусмотреть при выполнении работ по созданию информационной системы.

Так же агент Zabbix позволяет обеспечить мониторинг планируемых к использованию СУБД PostgeSQL (<https://www.zabbix.com/integrations/postgresql>), сервера приложений LiberCat (как Apache Tomcat, <https://www.zabbix.com/integrations/tomcat>) и Web-сервера Apache (<https://www.zabbix.com/integrations/apache>) путем использования соответствующих шаблонов для агента.

ASFAIK данная система мониторинга может использоваться и для мониторинга аппаратной части с использованием механизмов iLO и SMNP. Данный вопрос требует дополнительной проработки.

В обязательном порядке должна быть разработана и описана в документации на информационную систему стратегия проведения мониторинга ее работоспособности и производительности.

1. **Обеспечение безопасности**
2. **Сетевые протоколы**

В настоящее время мы обычно используем для информационных обменов незащищенные сетевые протоколы (HTTP, FTP, Telnet и пр.).

В дальнейшем мы должны использовать только защищенные сетевые протоколы (HTTPS, FTPS, SSH и пр.).

1. **Антивирусное программное обеспечение**

В настоящее время нами обеспечивается антивирусная защита только рабочих станций под управлением операционной системы Microsoft Windows.

В дальнейшем антивирусное программное обеспечение должно устанавливаться на всех рабочих станциях и серверах. При этом желательно обеспечивать централизованное обновление вирусных баз данных и централизованное управление антивирусным программным обеспечением путем подключения антивирусного программного обеспечения к средствам из состава системы обеспечения безопасности информации (СОБИ) ЦУП.

При этом следует отметить, что зачастую антивирусное программное обеспечение зачастую не очень хорошо функционирует на серверах баз данных, вызывая ложные срабатывания и блокировки. Многие поставщики СУБД прямо говорят о необходимости отключения антивирусного программного обеспечения на серверах баз данных, либо о необходимости его тонкой настройки с целью вывода исполняемых файлов, памяти и файлов данных СУБД из-под контроля антивирусного программного обеспечения. Это соображение необходимо учитывать при создании информационных систем.

В следующих письмах мы рассмотрим следующие вопросы:

1. **Используемые средства разработки**
2. **Организация процесса разработки**