With the Gemini conda channel unavailable due to the cyber incident, the packages you need for Gemini IRAF could be obtained from a colleague who had already installed Gemini IRAF prior to the cyber incident. Below we list the specific packages that you will need, and how to arrange them on your machine for conda to pick them up. Once you have that, the installation is straightforward and very similar to the standard method.

We will cover Mac Intel, Mac M1 and Linux.

IMPORTANT:

It is critical that you obtain packages that match your local machine architecture. For example, getting the packages from a colleague who uses Linux will do you no good if your machine is a Mac. For the Gemini IRAF VM, the Mac architecture Intel or M1 does matter. Be careful.

Step 1 - Be very nice to a colleague and obtain their packages.

The conda packages on your colleague's machine will be located in a directory equivalent to:

~/anaconda3/pkgs

(You can check the value of $CONDA_EXE and replace the "bin/conda" with "pkgs".)

The packages that you will need:

(If your colleague doesn't have them find a better colleague.)

Mac Intel/M1

- ds9-8.2.1-0.tar.bz2
- gemvm-1.0-py_0.tar.bz2
- libslirp-4.7.0-h254f028_1.tar.bz2 or -h8a0c2ca for M1
- qemu-7.0.0-hb3127f5_2.tar.bz2 or -h70deae4 for M1

Linux

- atlas-generic-3.10.2-1.tar.bz2
- ds9-8.2.1-0.tar.bz2
- iraf-2.16.UR.1-0.tar.bz2
- iraf-os-libs-1.0-0.tar.bz2
- iraf-x11-2.0-1.tar.bz2
- iraf.fitsutil-2011_02_22-2.tar.bz2

iraf.gemini-1.15-0.tar.bz2
iraf.stsdas-3.18.3-1.tar.bz2
iraf.tables-3.18.3-1.tar.bz2
pyraf-2.2.1-py310_2.tar.bz2

Step 2 - Thank your colleague!

Step 3 - Copy the packages in the necessary directory structure

For conda to correctly index the packages, it helps to put them into a certain directory structure.

Mac Intel
```
$ mkdir -p <somewhere>/condapkgs/osx-64
$ cp <all_the_bz2> <somewhere>/condapkgs/osx-64/
```

Mac M1
```
$ mkdir -p <somewhere>/condapkgs/noarch
$ mkdir -p <somewhere>/condapkgs/osx-64
$ mkdir -p <somewhere>/condapkgs/osx-arm64
$ cp gemvm-1.0-py_0.tar.bz2 <somewhere>/condapkgs/noarch/
$ cp ds9-8.2.1-0.tar.bz2 <somewhere>/condapkgs/osx-64/
$ cp libslirp* qemu* <somewhere>/condapkgs/osx-arm64/
```

Linux
```
$ mkdir -p <somewhere>/condapkgs/linux-64
$ cp <all_the_bz2> <somewhere>/condapkgs/linux-64
```

Step 4 - Fix the ~/.condarc

Open ~/.condarc in your favorite editor.

Make the "channels" section look like this (it's okay to comment out lines with a "#"):

```
channels:
  - conda-forge
```

There should be no stsci.edu or gemini.edu channels active.

Step 5 - Index your local packages

```
$ conda index <somewhere>/condapkgs
```

Step 6 - Set up the Gemini IRAF environment

Mac Intel

```
$ conda create -n gemvm_emergency -c <somewhere>/condapkgs/ gemvm
```

Mac M1

```
$ CONDA_SUBDIR=osx-arm64 conda create -n gemvm_emergency -c
<somewhere>/condapkgs/ gemvm
```

Linux

```
$ conda create -n geminidr_emergency -c <somewhere>/condapkgs/ python=3.10 iraf.gemini
ds9

$ conda activate geminidr_emergency
```

To test:

```
$ mkdir ~/iraf   (if not already set up)
$ cd ~/iraf
$ mkiraf        (if not already set up)
$ pyraf
--> gemini
--> gemtools
--> lpar gemcombine
```

Step 7 - For Mac, see the VM installation instructions

All we've done above is take care of the conda environment. To complete the installation, see the documentation:

https://gemini-iraf-vm-tutorial.readthedocs.io/en/stable/installation.html

---

Step 8 - Thank your colleague again!