```
from unittest import TestSuite, defaultTestLoader
from plone.app.blob.tests.base import BlobLinguaTestCase

from Products.CMFCore.utils import getToolByName
from plone.app.blob.tests.utils import getData, hasLinguaPlone
from plone.app.blob.tests.utils import makeFileUpload


class LinguaTests(BlobLinguaTestCase):

    def afterSetUp(self):
        self.setRoles(('Manager',))
        ltool = getToolByName(self.portal, 'portal_languages')
        ltool.manage_setLanguageSettings(defaultLanguage='en',
            supportedLanguages=('en', 'de'))

    def testCreateBlobelFish(self):
        guide = getData('plone.pdf')
        foo = self.folder[self.folder.invokeFactory('BlobelFish', 'flobby')]
        foo.update(title='Me fish.', guide=guide, language='en')
        # check content item
        self.assertEqual(foo.Title(), 'Me fish.')
        self.assertEqual(foo.Language(), 'en')
        self.assertEqual(foo.getPortalTypeName(), 'BlobelFish')
        self.assertEqual(foo.getContentType(), 'application/pdf')
        self.assertEqual(str(foo.getGuide()), guide)

    def testCreateTranslatedBlobelFish(self):
        guide = getData('plone.pdf')
        fish = self.folder[self.folder.invokeFactory('BlobelFish', 'flobby')]
        fish.update(title='Me fish.', guide=guide, language='en')
        # add a translation and check it
        fisch = fish.addTranslation('de', title='Ich Fisch.')
        self.assertEqual(fisch.Title(), 'Ich Fisch.')
        self.assertEqual(fisch.Language(), 'de')
        self.assertEqual(fisch.getPortalTypeName(), 'BlobelFish')
        self.assertEqual(fisch.getContentType(), 'application/pdf')
        self.assertEqual(str(fisch.getGuide()), guide)
        # as the field is language-independent both translations should
        # be using the same blob...
        self.assertEqual(fish.getGuide().getBlob(),
fisch.getGuide().getBlob())

    def testCreateAndRemoveTranslatedBlobelFish(self):
        guide = getData('plone.pdf')
        fish = self.folder[self.folder.invokeFactory('BlobelFish', 'flobby')]
        fish.update(title='Me fish.', guide=guide, language='en')
        # add a translation, after which both should use the same blob...
        fisch = fish.addTranslation('de', title='Ich Fisch.')
        self.assertEqual(fish.getGuide().getBlob(),
fisch.getGuide().getBlob())
        self.assertEqual(getattr(self.folder, 'flobby-de'), fisch)
        # now let's remove it again and make sure the blob's still okay...
        fish.removeTranslation('de')
        self.assertRaises(AttributeError, getattr, self.folder, 'flobby-de')
        self.assertEqual(str(fish.getGuide()), guide)

    def testEditTranslatedBlobelFish(self):
        guide = getData('plone.pdf')
        fish = self.folder[self.folder.invokeFactory('BlobelFish', 'flobby')]
        fish.update(title='Me fish.', guide=guide, language='en')
        # add a translation and then upload a new file...
        fisch = fish.addTranslation('de', title='Ich Fisch.')
        blob = fish.getGuide().getBlob()
        test = getData('test.pdf')
        fish.update(guide=makeFileUpload(test, 'test.pdf'))
        self.assertNotEqual(fish.getGuide().getBlob(), blob)
        self.assertEqual(fish.getGuide().getBlob(),
fisch.getGuide().getBlob())
        self.assertEqual(str(fish.getGuide()), test)
        self.assertEqual(str(fisch.getGuide()), test)


def test_suite():
    if hasLinguaPlone():
        return defaultTestLoader.loadTestsFromName(__name__)
    else:
        return TestSuite()
```